

## THESIS / THÈSE

### MASTER EN SCIENCES INFORMATIQUES

#### Mémoire Organisationnelle et Système d' Aide à la Négociation : ARBAS96

Mélard, Pierre; Vanreusel, Jean-François

*Award date:*  
1996

*Awarding institution:*  
Université de Namur

[Link to publication](#)

#### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

#### Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



Facultés Universitaires Notre-Dame de la Paix  
Institut d'Informatique  
rue Grandgagnage 21, 5000 NAMUR  
tél. (081) 72 49 83 - fax (081) 72 49 67

**Mémoire Organisationnelle et  
Système d'Aide à la Négociation:  
ARBAS96**

par Pierre Mélard et Jean-François Vanreusel

Promoteur: François Bodart

Mémoire présenté en vue de l'obtention du grade  
de Licencié et Maître en Informatique

Année académique 1995-1996

301357  
US 684785

## Résumé

Les recherches actuelles en informatique essaient de plus en plus d'améliorer la collaboration entre les individus et ceci qu'ils appartiennent à une même organisation ou à des organisations différentes. La négociation est l'une des activités primordiales de l'organisation, dans laquelle la collaboration et le travail de groupe jouent un rôle important. Le modèle d'argumentation ARBAS (*Action-Ressource Based Argumentation Support*) est le point de départ de notre recherche. Nous lui apportons différentes extensions pour qu'il réponde, d'une part, davantage aux besoins des systèmes d'aide à la négociation (*Negotiation Support Systems*) et, d'autre part, pour qu'il puisse gérer la mémoire organisationnelle restreinte aux processus de négociation. Cette mémoire organisationnelle met à la disposition de l'organisation et des négociateurs une quantité importante d'informations sur les anciennes négociations, les acteurs, leurs buts, leur histoire, ... . Ces informations peuvent notamment servir à de l'apprentissage organisationnel.

Mots-clés: Négociation, NSS, Mémoire Organisationnelle, Apprentissage, Argumentation.

## Abstract

A lot of the actual effort in computer science has been focused to improve the way people collaborate within an organization or among several organizations. Negotiation is one of the crucial activities of organizations in which collaboration and workgroup play an important role. We use the argumentation model ARBAS (*Action-Ressource Based Argumentation Support*) as a fundation for our work. We extend it in several directions for, first, meeting the needs of Negotiation Support Systems more, and then, for managing organizational memory limited to negotiation processes. This organizational memory supply negotiators and organizations with a lot of information about previous negotiations, actors and their goals, actions, ... . This information can especially be used for organizational learning

Keywords: Negotiation, NSS, Organizational Memory, Organizational Learning, Argumentation.





## **Remerciements**

La réalisation d'un stage à l'étranger et la rédaction d'un mémoire n'est pas une chose aisée. Nous tenons à remercier toutes les personnes qui, d'une manière ou d'une autre, nous ont aidé à réaliser notre mémoire.

Nous remercions, tout particulièrement, les professeurs F. Bodart et T. Bui pour la formidable expérience qu'ils nous ont amenés à vivre. L'introduction dans le monde de la recherche fût pour nous une aventure particulièrement enrichissante. Ils se sont toujours montrés disponibles malgré leur emploi du temps très chargé.

Nos remerciements vont aussi aux membres du corps professoral de l'Institut pour leurs précieux conseils et leur disponibilité durant toutes nos études.



## Préambule

De manière à mettre en pratique les connaissances que nous avons acquises au cours de nos études en informatique, nous avons effectué un stage d'une durée de quatre mois à la Naval Postgraduate School de Monterey (Californie -USA). La Naval Postgraduate School est une école de haut niveau qui a pour objectif de former l'élite de l'armée américaine dans le but de lui pourvoir des postes à responsabilités élevées. Une abondance de domaines de recherche sont représentés au sein de cette école : océanographie, météorologie, astronomie, informatique, mathématiques, chimie, etc... . Pour montrer à quel point cette école est prestigieuse, nous citerons juste qu'un tiers des astronautes des navettes spatiales américaines y ont été formés (l'école dispose de son propre simulateur) et que l'école est peut-être l'une des seules institutions d'enseignement dans le monde à disposer de son propre satellite dans l'espace.

Si nous avons eu la chance de travailler dans cet environnement exceptionnel, nous le devons au professeur Tung Bui. Le professeur Tung Bui, docteur en sciences économiques ainsi qu'en informatique, peut être reconnu comme un spécialiste dans le large domaine des systèmes d'aide à la décision. Il a participé à l'élaboration de différents systèmes (Co-op, Negotiator, ARBAS, .... ). Le professeur Tung Bui nous a invités à effectuer notre stage au sein du département de *Management Information System* de la Naval Postgraduate School. Le projet auquel nous avons été conviés à participer est le projet ARBAS qui nous a permis de nous familiariser avec le monde des systèmes d'aide à la négociation. Le Professeur Tung Bui nous a demandé d'analyser ce système et de voir quelles extensions il serait possible d'y apporter. Nos recherches ont concerné ces extensions et nous les exposons dans ce travail.

Outre le cadre unique dans lequel nous étions conviés à travailler, nous avons particulièrement été séduits par le monde de la recherche auquel nous n'étions guère habitués. Nous avons aussi été étonnés par la grande liberté d'actions que notre hôte nous a témoignée. Tout en nous lançant quelques pistes de recherche, il nous a toujours laissé mener notre étude dans la direction qui nous semblait être la meilleure.

L'expérience que nous avons eu le bonheur de vivre sous la guidance du Professeur Tung Bui fût pour nous une aventure exceptionnelle de laquelle nous sommes certainement sortis grandis.





---

## **Table des matières**

Remerciements	5
Préambule	7
<b>Table des matières</b>	<b>9</b>
<b>Introduction</b>	<b>11</b>
<b>PARTIE I : La mémoire organisationnelle</b>	<b>13</b>
<b>I.1 Introduction</b>	<b>13</b>
<b>I.2 De la mémoire individuelle à la mémoire organisationnelle</b>	<b>13</b>
<b>I.3. Définition de la mémoire organisationnelle</b>	<b>16</b>
<b>I.4. Rôles et utilité de la mémoire organisationnelle</b>	<b>17</b>
<b>I.5. L'apprentissage organisationnel</b>	<b>20</b>
I.5.1. Définitions	20
I.5.2. Types d'apprentissages organisationnels	20
I.5.3. Les processus d'apprentissage organisationnel	21
<b>I.6. Les dangers de la mémoire organisationnelle</b>	<b>22</b>
<b>I.7. Modèles de mémoire organisationnelle</b>	<b>24</b>
I.7.1. Modèle de Walsh et Ungson	24
I.7.2. Modèle de Stein et Zwass	27
I.7.3. Modèle de Morrison et Weiser	29
<b>I.8. Les systèmes supportant la mémoire organisationnelle</b>	<b>33</b>
I.8.1. Pourquoi développer les SISMOs ?	33
I.8.2. Comment utiliser les SISMOs ?	34
I.8.3. Les qualités d'un SISMO	34
I.8.4. Les problèmes des SISMOs	37
<b>I.9. Exemples de SISMOs</b>	<b>39</b>
I.9.1. Les systèmes d'intégration de documents	39
I.9.2. Les systèmes de support à l'activité humaine	39
I.9.3. Les systèmes de capture et d'assimilation de connaissances	42
I.9.4. Les SISMOs qui surgissent	42
<b>I.10. Critiques du concept et des modèles de mémoire organisationnelle</b>	<b>45</b>
<b>I.11. Résumé</b>	<b>51</b>

<b>PARTIE II : La mémoire organisationnelle dans le cadre de la négociation</b>	<b>53</b>
<b>II.1. Introduction</b>	<b>53</b>
<b>II.2. Etat de l'art des systèmes d'aide à la négociation</b>	<b>53</b>
<b>II.3. Pourquoi mémoriser les processus de négociation ?</b>	<b>61</b>
<b>II.3. Présentation générale d'ARBAS</b>	<b>68</b>
II.3.1. Cadre d'utilisation d'ARBAS	68
II.3.2. Les règles d'accès d'ARBAS	69
II.3.3. Architecture générique d'ARBAS	70
<b>II.4. ARBAS - outil d'aide à la négociation</b>	<b>71</b>
II.4.1. La lexicographie d'ARBAS	71
II.4.2. La syntaxe d'ARBAS	72
II.4.3. La sémantique d'ARBAS	73
II.4.4. Le schéma graphique d'ARBAS	74
II.4.5. Exemple	75
II.4.6. Les limites et les extensions d'ARBAS	79
II.4.7. ARBAS étendu dans le cadre de la négociation	85
<b>II.5. ARBAS96 en tant que SISMO</b>	<b>85</b>
II.5.1. Extension d'ARBAS en termes de concepts	86
II.5.2. Schéma Entité-Association d'ARBAS96	90
II.5.3. ARBAS96 vs ARBAS	102
II.5.4. Extension d'ARBAS en termes de fonctionnalités	103
II.5.5. Le fonctionnement général d'ARBAS96	105
II.5.6. Mise en œuvre d'ARBAS96	115
II.5.7. Implémentation d'ARBAS96	135
II.5.8. Etude de cas : Softwood Lumber	138
II.5.9. ARBAS96 est un SISMO	151
II.5.10. Limites d'ARBAS96 et vers ARBAS97	154
<b>II.6. Résumé</b>	<b>156</b>
<b>Conclusion</b>	<b>157</b>
<b>Bibliographie</b>	<b>159</b>



---

## Introduction

Au cours de la dernière décennie, de nombreux progrès ont été constatés dans le domaine du partage des informations dans l'espace. L'explosion d'Internet en constitue certainement une preuve. Dernièrement, les recherches se sont orientées vers le transfert de l'information, non plus à travers l'espace, mais à travers le temps. Cette nouvelle idée consiste à créer une mémoire pour l'organisation, contenant toutes les informations passées ou présentes, internes ou externes, relatives à son activité et qui peuvent servir de support aux processus organisationnels.

Un autre champ qui s'est aussi particulièrement bien développé au cours de ces dernières années est le domaine des systèmes d'aide à la négociation. Ces systèmes appartiennent à la famille des systèmes d'information d'aide à la décision. Leur objectif est de soutenir, faciliter les processus de résolution de conflits dans lesquels plusieurs parties peuvent être impliquées.

Dans ce travail, nous nous proposons d'étudier la relation entre la mémoire organisationnelle et les systèmes d'aide à la négociation. Dans une première partie, nous présentons les différentes idées qui ont été émises en matière de mémoire organisationnelle, tant au niveau général qu'au niveau des systèmes informatiques qui peuvent la supporter. Cette présentation concerne essentiellement trois modèles de mémoire organisationnelle établis, successivement, par Walsh et Ungson (1991), Stein et Zwass (1995) et Morrison et Weiser (1996). Nous exposons aussi, à titre d'exemples, quelques systèmes qui supportent la mémoire organisationnelle. Ces exemples mettent en évidence le fait que, pour l'instant, aucun système ne répond aux exigences développées dans les modèles théoriques de mémoire organisationnelle.

Dans la deuxième partie de ce travail, nous étudions le monde des systèmes d'aide à la négociation et nous lui appliquons les idées émises en matière de mémoire organisationnelle. La base de notre recherche consiste à étendre le langage d'argumentation ARBAS (*Action-Resource Based Argumentation Support*) (Binbasioglu et al. (1995) et Bui et al. (1995)). Ce langage a initialement été conçu pour servir de moyen de communication entre les participants d'une négociation. Dans un premier temps, nous améliorons ARBAS pour qu'il réponde davantage aux exigences des négociations. Ensuite, nous l'étendons sur base de différentes exigences présentées dans les modèles de mémoire or-

ganisationnelle. Cette deuxième extension est à l'origine de la nouvelle version d'ARBAS, intitulée ARBAS96.

L'intérêt de l'extension que nous menons en terme de mémoire organisationnelle réside surtout dans le fait que, pour l'instant, les modèles liés à ce concept sont encore vagues. Ils ne définissent pas encore d'une manière détaillée quelles sont les données et fonctions qui doivent être présentes dans les systèmes d'information qui supportent la mémoire organisationnelle. L'objectif principal de notre travail consiste à apporter des éléments de réponse par rapport à ces faiblesses, tout en nous limitant au cadre des systèmes d'information d'aide à la négociation.



---

## ***PARTIE I : La mémoire organisationnelle***

---

### ***1.1 Introduction***

Dans cette partie, nous proposons un aperçu des différentes idées émises en matière de mémoire organisationnelle. Dans un premier temps, nous présentons des concepts fondamentaux liés à la mémoire de l'homme et "translatés" vers la mémoire de l'organisation. Nous proposons différentes définitions qui ont été énoncées dans la littérature et, par rapport auxquelles, nous soulevons quelques critiques. Ensuite, nous exposons les différents rôles assumés par la mémoire organisationnelle en insistant particulièrement sur l'un d'entre-eux : l'apprentissage organisationnel. La maintenance d'une mémoire organisationnelle ne se fait pas sans un certain nombre de dangers que nous exposons aussi dans cette partie. Enfin, nous détaillons trois modèles liés à la mémoire organisationnelle et particulièrement populaires dans la littérature. Nous présentons aussi quelques systèmes supportant la mémoire organisationnelle. De manière à être précis, nous utiliserons plutôt l'expression "systèmes supportant de la mémoire organisationnelle" dans la mesure où, pour l'instant, aucun système ne la supporte entièrement. Nous clôturons cette partie par une critique de ce concept et des modèles qui y sont liés.

---

### ***1.2 De la mémoire individuelle à la mémoire organisationnelle***

D'une manière générale, la mémoire peut se définir comme étant une activité psychique qui assure la fixation des événements et l'évocation du passé (Encyclopédie Atlas). Telle qu'elle est énoncée, cette définition cible l'individu comme d'ailleurs la plupart des définitions relatives à la mémoire lorsque celle-ci ne concerne pas l'informatique, évidemment. De nombreuses études ont été menées pour essayer de déterminer le fonctionnement de la mémoire de l'homme. Dans le cadre de ce travail, nous nous contenterons de rappeler quelques principes fondamentaux développés dans la littérature (Tuomi (1995), Sougné (1994)).

La mémoire à long terme de l'individu est composée de deux parties distinctes reposant sur des systèmes cérébraux différents : la **mémoire déclarative** et la **mémoire non déclarative** (voir Tableau I.1) (Sougné (1994)). La mémoire déclarative est une mémoire consciente des faits et des événements. Elle est composée de la **mémoire épisodique** et de la **mémoire sémantique** (Sougné (1994), Tuomi (1995)). La mémoire épisodique est



la mémoire autobiographique des faits, des événements -des épisodes, en somme - occupant une place particulière dans le temps et dans l'espace. Le souvenir d'une "première fois" fait, par exemple, appel à la mémoire épisodique. La mémoire sémantique, quant à elle, se rapporte aux connaissances générales du monde (Sougné (1994)). Elle permet, par exemple, à l'individu de reconnaître les objets et personnes qui l'entourent. Enfin, la mémoire non déclarative est la mémoire non consciente des événements comprenant les habitudes et habiletés (e.g. la conduite d'une voiture, les automatismes en sports), ainsi que le conditionnement (Sougné (1994)).

Mémoires déclaratives	Mémoire épisodique	La mémoire des événements. Quand ? Où ?
	Mémoire sémantique	Les connaissances générales du monde
Mémoires non déclaratives	Habiletés et habitudes,  Conditionnement	

Tableau I.1 : Les mémoires à long terme - Adapté de Sougné (1994)

Tuomi (1995) décrit le processus de mémorisation en insistant sur l'aspect évolutif et dynamique d'un tel processus (voir Figure I.1.). Le processus associé à la mémoire sémantique est l'**abstraction** et peut être considéré comme une forme d'oubli, de désapprentissage. La **perception** est le processus associé à la mémoire épisodique. Il s'agit d'un mécanisme qui capte des événements dans le monde perceptuel constitué d'expériences qui n'ont pas encore été vécues, assimilées.

Le processus de mémorisation, selon Tuomi (1995), doit être vu comme un processus d'accumulation accompagné d'un processus de désapprentissage dans lequel les détails des épisodes perdent leur pertinence lorsque survient l'abstraction. Le caractère dynamique de ce processus devient visible seulement lorsqu'il est compris comme un processus évolutif où les abstractions et les perceptions sont liées à l'histoire de l'individu.

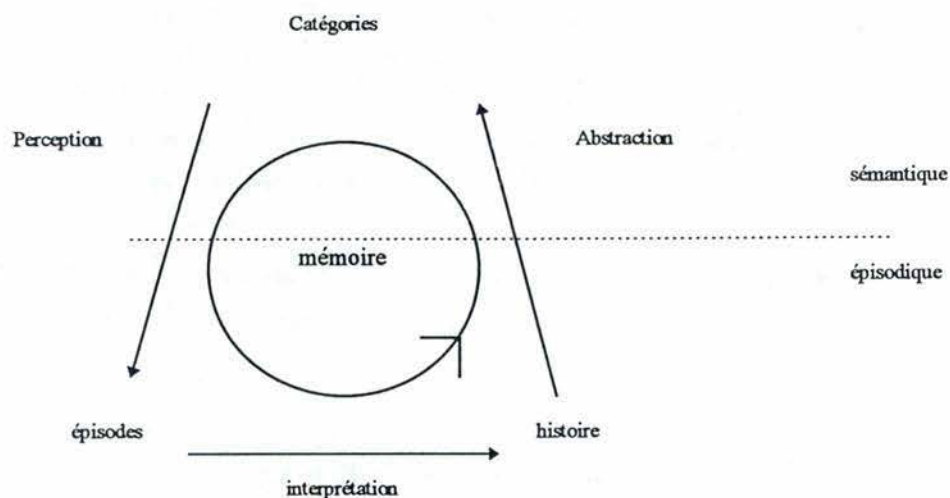


Figure 1.1. : Les processus de mémorisation - Adapté de Tuomi (1995)

Les recherches en matière de mémoire n'ont pas uniquement été portées au niveau de l'individu puisque des chercheurs se sont aussi intéressés à la mémoire collective et en particulier à la mémoire des organisations, plus communément appelée **mémoire organisationnelle**. Le passage de la mémoire individuelle à la mémoire organisationnelle ne se fait pas sans difficultés. Les chercheurs ne sont pas encore d'accord sur la forme spécifique de la mémoire organisationnelle et sur le rôle qu'elle devrait assumer dans l'organisation (Walsh et Ungson (1991)).

Le fait de considérer que les organisations disposent d'une mémoire soulève le problème de l'**anthropomorphisme**. A ce propos, Walsh et Ungson (1991) rappellent que les théories prétendant que des organisations pouvant apprendre, donner naissance, reproduire ou encore mourir, ont souvent été l'objet de critiques. Par rapport à ces critiques, nous pensons que la métaphore offre une manière didactique pour expliquer le fonctionnement de la mémoire organisationnelle, ce qui rejoint l'idée d'Ackerman (1994) qui considère l'anthropomorphisme comme un très bon moyen pour comprendre les systèmes vivants comme les organisations.

Nous n'oserons évidemment pas prétendre qu'il est possible de créer une mémoire organisationnelle qui fonctionne exactement comme celle de l'homme mais, nous pensons toutefois que des caractéristiques relatives à la mémoire humaine peuvent aussi être présentes dans celle de l'organisation. A ce propos, il apparaît que la mémoire organisationnelle contient aussi bien de la mémoire épisodique que de la mémoire sémantique (El Sawy et al. (1986)). La mémoire sémantique d'une organisation contient des informations



sur les objets, les concepts (e.g. une facture, un client) manipulés par l'organisation, tandis que la mémoire épisodique enregistre des faits, des événements, des processus qui se déroulent dans l'organisation (e.g. prise de décision, processus de négociation). Actuellement, les organisations conservent surtout de la mémoire sémantique. Par contre, par rapport à la mémoire épisodique, les organisations enregistrent bien les faits, les finalités de ces faits mais, généralement, peu d'attention est portée à la manière dont ces faits ont été établis. Dans le cas d'une prise de décision, par exemple, les organisations ont plutôt tendance à n'enregistrer que son résultat, plutôt que toutes les étapes qui ont permis d'y arriver.

### ***1.3. Définition de la mémoire organisationnelle***

---

Plusieurs définitions de la mémoire organisationnelle ont déjà été présentées dans la littérature. La plus populaire d'entre elles est certainement celle de Walsh et Ungson (1991) :

*"Organisational memory refers to stored information from an organization's history that can be brought to bear on present decisions."*

Watson (1996) présente une définition quelque peu étendue par rapport à celle de Walsh et Ungson :

*"Organizational memory is an organization's record of the details necessary for transacting business and making decisions."*

Cette tendance à augmenter le terrain d'action de la mémoire organisationnelle se retrouve aussi dans une définition basée sur l'efficacité organisationnelle et énoncée par Stein et Zwass (1995) :

*" We consider organizational memory to be the means by which knowledge from the past is brought to bear on present activities, thus resulting in higher or lower levels of organizational effectiveness."*

Pour Stein et Zwass, la connaissance ne sert pas seulement de support aux décisions mais à toutes les activités de l'organisation (prises de décision, organisation, conception, communication, planification, négociation, ...). Nous reprendrons cette définition pour notre travail et nous l'énoncerons en français comme suit :

La mémoire organisationnelle est l'ensemble des moyens par lesquels les informations du passé peuvent être amenées à supporter les activités présentes de l'organisation, résultant donc en des niveaux d'efficacité organisationnelle inférieurs ou supérieurs.

Les définitions énoncées dans cette section sont très générales et, au moment où nous présenterons les modèles de mémoire organisationnelle, nous expliquerons ce qu'il faut comprendre par *ensemble d'informations* ainsi que les différentes fonctions nécessaires à la gestion de cet ensemble.

#### ***1.4. Rôles et utilité de la mémoire organisationnelle***

---

Après avoir défini la mémoire organisationnelle, il est intéressant de se pencher sur ses rôles et son utilité. En termes d'introduction, nous relevons certains avantages qui ont été émis à propos de la mémoire humaine.

Lorsqu'elle concerne les êtres vivants, la mémoire semble être nécessaire pour plusieurs fonctions relevées par Tuomi (1995), à savoir : **l'adaptation** (Wiener (1961), Kohonen (1984)), pour la **reconnaissance de formes et l'organisation de soi-même** (Haken (1983), Nicolis et Prigogine (1977)), la **catégorisation** (Pao (1989), Touretzky (1989)) et **l'anticipation** (Rosen, (1985)). Nous ajouterons que la mémoire permet aussi d'**améliorer la compétence** (Hedberg (1981)) et d'**augmenter l'autonomie** des individus (Churchman et al. (1957)). Dans le domaine plus restreint de l'argumentation et de la négociation, Tuomi (1995) considère la mémoire comme un **besoin fondamental pour la communication**.

Walsh et Ungson (1991) ont relevé trois rôles principaux joués par la mémoire organisationnelle: un rôle informationnel, un rôle de contrôle et un rôle de pouvoir. Chacun de ces rôles regroupe un certain nombre d'utilités de la mémoire organisationnelle.

##### **Le rôle informationnel**

Le rôle informationnel est lié au partage de l'information dans le temps et dans l'espace. L'information étant une forme de pouvoir, son partage ne se fait pas sans une certaine réticence de la part des membres de l'organisation et, c'est pourquoi Walsh et Ungson préconisent un système de récompenses et de punitions pour les bons et les mauvais élèves.

Dans ce rôle informationnel, la mémoire organisationnelle :



- permet de réduire le phénomène de "réinvention de la roue" (Rao et Goldman-Segall (1995)). Au cours de son histoire, une organisation doit faire face à de nombreux problèmes. Il est intéressant, pour l'organisation, de mémoriser les informations relatives aux résolutions de ces problèmes car, dans le cas où elle doit à nouveau y faire face, elle se souviendra des solutions qu'elle y avait appliquées et ne devra donc pas se relancer dans un processus de recherche de solution. La mémoire organisationnelle offre, dans ce cas, une épargne de ressources. Dans le cadre plus restreint des réunions de travail, les humains se souviennent difficilement du passé et plus particulièrement de la raison de leurs décisions (Schwabe (1994)). Le stockage d'informations peut remédier à cette lacune.
- aide à éviter que des erreurs passées se reproduisent en explorant les solutions de problèmes similaires. Une organisation qui ne commet pas d'erreurs, si petites soient-elles, cela n'existe pas. La mémorisation d'informations relatives aux résolutions de problèmes permet non seulement de voir quelle solution a été appliquée à un problème (voir point précédent) mais offre aussi un moyen d'évaluation (feed-back) de ces solutions. Le feed-back permet de rejeter les solutions qui se sont, en fin de compte, avérées inefficaces ou même néfastes pour l'organisation.
- offre la possibilité d'éduquer, d'informer le nouveau personnel des problèmes historiques. Lorsqu'un nouveau membre du personnel entre en fonction, son ignorance en matière des problèmes auxquels l'organisation a dû faire face est grande. La mémoire organisationnelle lui permet de mieux comprendre ces difficultés ainsi que la manière qui a été utilisée pour les contourner.
- améliore les définitions de problèmes. Etant donné que les membres de l'organisation disposent de plus d'informations, ils peuvent élaborer des définitions de problème plus complètes, ce qui les rend plus solides. De plus, ils peuvent aussi se servir des définitions de problèmes passés pour améliorer les définitions présentes.



- améliore les plannings et la stratégie future. Etant donné que la mémoire organisationnelle enregistre l'activité d'une organisation, il est possible de récupérer les plannings passés pour en tirer des leçons. Les analyses posées sur ces plannings permettent de mesurer leur degré d'efficacité et peuvent, dès lors, servir de support aux élaborations actuelles de planning. La stratégie est aussi améliorée grâce à l'expérience acquise par l'organisation. La mémoire organisationnelle peut offrir, dans certaines limites, des conseils sur la mise en place des stratégies et des prévisions sur l'évolution de ces stratégies.

### **Le rôle de contrôle**

Le rôle de contrôle est lié à l'observation des activités présentes. Il s'agit de vérifier que les activités se déroulent bien comme prévu.

Dans ce rôle de contrôle, la mémoire organisationnelle :

- permet de diminuer les coûts de transaction qui sont souvent associés à l'implémentation d'une nouvelle décision. Le contrôle de cette implémentation doit assurer que les erreurs passées ne se reproduisent pas.

### **Le rôle de pouvoir**

La mémoire organisationnelle assure un rôle politique. L'information est détenue par des individus qui jouent les figures centrales de l'organisation étant données les informations qu'ils détiennent. La possession d'informations crée une source de dépendance par laquelle des individus ou des groupes sont capables d'influencer les actions d'autrui.

Dans ce rôle de pouvoir, la mémoire organisationnelle :

- permet de diminuer la dépendance de l'organisation vis-à-vis de certains de ses membres qui disposent d'un accès privilégié à l'information. Lorsqu'un membre a besoin d'informations pour remplir l'une de ses tâches, il faut qu'il y ait accès sans devoir passer par un intermédiaire. Il faut cependant noter que cet accès ne doit pas concerner toutes les informations de l'organisation. A chaque membre correspond un domaine d'informations accessibles défini en fonction des tâches qui lui sont assignées.

Un problème important qui occupe les organisations est la **rotation du personnel**. Lorsque des membres quittent l'organisation, ils emportent avec eux une partie de la mémoire organisationnelle. Face à ce problème, Tuomi (1995) définit la **mémoire institutionnelle** comme étant les composants de la mémoire organisationnelle qui peuvent être considérés comme indépendants des membres de l'organisation (e.g. procédures standards, bases de données des transactions, etc...). Autrement dit, si l'échange de deux membres quelconques de l'organisation laisse la mémoire intacte, cette mémoire est purement institutionnelle. Afin de résoudre l'amnésie due aux rotations de personnel, les organisations devront s'arranger pour avoir une mémoire institutionnelle la plus grande possible ce qui les rendront moins dépendantes de leurs membres. Dans cet objectif, les rôles informationnel et de pouvoir joués par la mémoire organisationnelle sont très importants.

### ***1.5. L'apprentissage organisationnel***

---

La plupart des rôles de la mémoire organisationnelle, présentés dans la partie précédente sont liés à l'apprentissage. Il s'agit essentiellement de se servir des informations du passé pour améliorer les processus présents.

#### ***1.5.1. Définitions***

Plusieurs définitions ont été proposées pour l'apprentissage organisationnel. Comme pour la mémoire organisationnelle, ces définitions sont récentes puisqu'elles datent de moins de vingt ans. En 1978, Argyris et Schon considéraient l'apprentissage organisationnel comme un processus de détection et de correction d'erreurs. Plus tard, Fiol et Lyles (1985) ont défini l'apprentissage comme étant un processus améliorant les actions grâce à de meilleures connaissances et compréhensions.

Enfin, Dogson (1993) considère l'apprentissage organisationnel comme la manière dont les entreprises construisent, incrémentent et organisent la connaissance et les routines autour de leurs activités et au sein de leur culture, ainsi qu'elles adaptent et développent l'efficacité organisationnelle en améliorant les aptitudes des employés.

#### ***1.5.2. Types d'apprentissages organisationnels***

Argyris et Schon (1978) décrivent trois types d'apprentissage organisationnel :



- **L'apprentissage en simple boucle (*single-loop learning*)** : cet apprentissage apparaît lorsque des erreurs sont détectées et corrigées mais que les entreprises ne modifient pas leurs politiques et buts actuels. Il s'agit d'un apprentissage de bas niveau. Il y a apprentissage mais celui-ci n'altère pas la nature fondamentale des activités de l'organisation.
- **L'apprentissage en double boucle (*double-loop learning*)** : cet apprentissage se produit lorsque, en plus d'une détection et d'une correction des erreurs, l'organisation apporte des modifications à ses normes, procédures, politiques et objectifs. Il s'agit d'un apprentissage de haut niveau, également appelé apprentissage stratégique.
- **L'apprentissage Deutero (*Deutero-learning*)** : cet apprentissage a lieu lorsque les organisations savent comment mener à bien l'apprentissage en simple et en double boucle. Les deux premiers types d'apprentissage ne se produiront pas si les organisations ne sont pas conscientes que l'apprentissage doit se dérouler. La prise de conscience du besoin d'apprentissage permet de proposer des orientations et des styles d'apprentissage.

Nous dirons encore que l'apprentissage peut intervenir, peu importe son type, dans différentes fonctions de l'organisation telles que la recherche, le développement, la conception, l'ingénierie, la manutention, le marketing, l'administration et les ventes.

### ***1.5.3. Les processus d'apprentissage organisationnel***

Huber (1991) considère que l'apprentissage organisationnel est basé sur les quatre constructions suivantes :

- **Acquisition de la connaissance** : l'apprentissage se produit lorsque l'organisation acquière de la connaissance. L'acquisition de la connaissance est effectuée en surveillant l'environnement, en utilisant des systèmes d'information pour stocker, gérer et récupérer les informations, en faisant de la recherche et du développement, en suivant des formations, en surveillant des brevets, en consultant les bibliothèques. L'apprentissage n'est pas uniquement dû à l'acquisition de connaissances extérieures à l'organisation, mais dépend aussi de la réorganisation des connaissances existantes.

- **Distribution des informations** : la distribution des informations consiste à partager des informations entre les unités et les membres d'une organisation. Les connaissances relatives aux savoir-faire, lettres, mémos, conversations informelles et rapports sont capturées et diffusées au sein de l'organisation. La plupart du temps, l'apprentissage organisationnel trouve son origine dans des anecdotes ou des histoires liées à la pratique du travail plutôt que dans des manuels ou autres documents formels. Au plus le partage d'information est grand, au plus l'apprentissage organisationnel est important.
- **Interprétation des informations** : l'interprétation des informations est un processus qui attache une ou plusieurs significations aux informations distribuées. Huber (1991) prétend que les individus et groupes sont munis de structures innées de croyances qui leurs permettent d'interpréter les informations et donc d'y apporter une signification. Plusieurs interprétations peuvent être posées sur une même situation. Tout le monde ne voit pas toujours les choses sous le même angle, mais il faut pourtant que l'apprentissage soit efficace pour tous. Dès lors, au plus le nombre d'interprétations différentes est important, au plus l'apprentissage est grand.
- **Mémoire organisationnelle** : la mémoire organisationnelle est le référentiel (*repository*) dans lequel la connaissance est stockée pour un usage ultérieur. Selon Huber (1991), la mémoire organisationnelle joue un rôle critique dans l'apprentissage organisationnel. L'utilisation de l'apprentissage organisationnel dépend de l'efficacité de la mémoire organisationnelle.

### ***1.6. Les dangers de la mémoire organisationnelle***

---

Comme nous l'avons présenté plus haut, la mémoire et, en particulier la mémoire organisationnelle, présente de nombreux avantages. Cependant, de manière à être complets, il nous faut émettre, non pas des désavantages, mais des dangers d'utilisation de la mémoire et de l'apprentissage organisationnel.

Walsh et Ungson (1991) soulèvent le problème de l'**inflexibilité** qui peut être causé par la mémoire organisationnelle. Face à une situation difficile, les membres de l'organisation recherchent des situations semblables dans l'histoire de l'organisation et appliquent la solution de l'époque au problème présent. L'inflexibilité trouve son origine dans l'apprentissage en simple-boucle qui entraîne l'application constante des mêmes solutions. De plus, les membres ne réfléchissent plus eux-mêmes aux solutions des problè-



mes auxquels ils sont confrontés mais appliquent automatiquement les solutions proposées alors qu'ils auraient peut-être pu en trouver une autre eux-mêmes. L'esprit de créativité des membres de l'organisation se voit dès lors limité. Lorsque les membres appliquent aveuglément une solution d'un problème passé, cela peut aussi mener l'organisation dans les ennuis du fait que le contexte présent est peut-être tout à fait différent de celui dans lequel la solution avait été proposée. Le danger d'appliquer une solution d'un problème passé à un problème présent réside essentiellement dans le degré d'obsolescence de cette solution.

La **mémorisation du contexte** dans lequel s'est déroulée une situation est essentielle. Sans contexte, il est difficile, voire impossible, de percevoir les conditions dans lesquelles une décision, par exemple, a été prise. Le contexte est un élément crucial pour l'interprétation d'une situation mémorisée. Ackerman (1994) souligne le fait que l'interprétation d'une situation sera d'autant plus facile que la mémorisation de cette situation est récente, car une partie importante du contexte sera toujours d'actualité. Le problème de la gestion du contexte a poussé Ackerman à distinguer la mémoire de groupe de la mémoire organisationnelle. Dans une mémoire de groupe, un contexte existe déjà puisque les membres du groupe partagent le même monde linguistique et social. Par contre, dans une mémoire organisationnelle, il faut faire appel à plus de formalisation pour permettre la compréhension de tous. Le degré de formalisation dépend de l'usage de la mémoire organisationnelle. Si cet usage est limité, l'information peut être moins formalisée. Remarquons aussi que lorsqu'il s'agit de mémoriser de l'information à long terme, la mémoire de groupe sera beaucoup plus formalisée.

Deux autres aspects dangereux nécessitant quelques précautions est le **respect de la vie privée et de la confidentialité des informations** (Schwabe (1994)). Lors d'une réunion, des échanges verbaux s'opèrent, certains d'entre-eux, de part leur nature confidentielle, ne doivent pas être repris dans le compte-rendu de la réunion. De plus, les informations échangées, au cours de réunions, peuvent concerner toute une organisation ou seulement une partie de celle-ci. Il s'impose donc une réglementation de l'accès à l'information. Nous ajouterons encore qu'après une réunion, certains points doivent tout simplement être oubliés.

Un autre point critique soulevé par Walsh et Ungson (1991) est le **gaspillage de ressources** (temps et argent) que l'utilisateur doit consacrer pour récupérer les informations du passé. Il est, en effet, impensable que la recherche, au sein de la mémoire organisationnelle, d'une solution d'un problème passé nécessite davantage de temps que



l'élaboration d'une nouvelle solution. A ce niveau, les systèmes d'information peuvent jouer un rôle important.

## ***1.7. Modèles de mémoire organisationnelle***

---

L'intérêt que les scientifiques ont porté à la mémoire organisationnelle est récent et peu de modèles ont été proposés jusqu'à présent. Dans cette section, nous présentons d'abord le modèle de Walsh et Ungson (1991) qui a été référencé à de nombreuses reprises dans la littérature. Ce modèle est solide mais fait preuve de faiblesses lorsqu'il s'agit d'étudier la relation qui peut exister entre la mémoire organisationnelle et les systèmes d'information. Dans le but de remédier à cette lacune, Stein et Zwass (1995) ont proposé un modèle de Système d'Information Supportant la Mémoire Organisationnelle (ou SISMO). En conclusion de cette section, nous exposons le modèle de Morrison et Weiser (1996) qui assure, en quelque sorte, l'unification des deux premiers modèles.

### ***1.7.1. Modèle de Walsh et Ungson***

Dans leur modèle de mémoire organisationnelle, Walsh et Ungson proposent 3 fonctions assurant la gestion de la mémoire : l'acquisition, la rétention et la récupération.

#### **L'acquisition**

Cette fonction a pour but de capturer des informations relatives aux prises de décision et aux problèmes résolus.

#### **La rétention**

Cette fonction est liée au stockage des informations internes et externes captées par l'organisation. Walsh et Ungson présentent six « coffrets » de stockage :

- **Les individus**  
Les individus ont leur propre souvenir de ce qui s'est passé dans et à propos d'organisations. Ce souvenir se base sur leur propre expérience ainsi que sur des observations. Les individus peuvent se servir de fichiers et d'enregistrements comme aide-mémoire.
- **La culture**

La culture est la manière apprise de percevoir, penser, ressentir les problèmes qui sont transmissibles aux membres de l'organisation (Schein, 1984). La culture est présente dans le langage, les cadres de travail partagé, les symboles, les histoires, les sagas et les grappes.

- **Les transformations**

Une transformation est une conversion d'un input (e.g. matières premières) en un output (e.g. produit fini). L'information est présente dans de nombreuses transformations qui se produisent dans les organisations et au travers de celles-ci. La mémoire est préservée dans une variété de procédures, de règles et de systèmes formalisés.

- **Les structures**

Dans les organisations, il se présente différents rôles dans lesquels il est possible de stocker de l'information. A chaque rôle, il est possible de porter une prévision par rapport au comportement. Le rôle d'un médecin est, par exemple, prévisible.

- **L'écologie (*ecology*)**

L'aspect physique d'une organisation influence le comportement de ceux qui y travaillent ainsi que la relation entre les employés et les visiteurs. De ce fait, l'écologie retient de l'information contextuelle à propos d'une organisation et de ses membres.

- **Les archives externes**

Une organisation ne détient pas à elle seule les connaissances sur son passé. Celles-ci sont aussi détenues par les concurrents, les gouvernements, les entreprises de services financiers, les marchés électroniques, les médias (télévision, presse écrite) et les historiens de gestion.



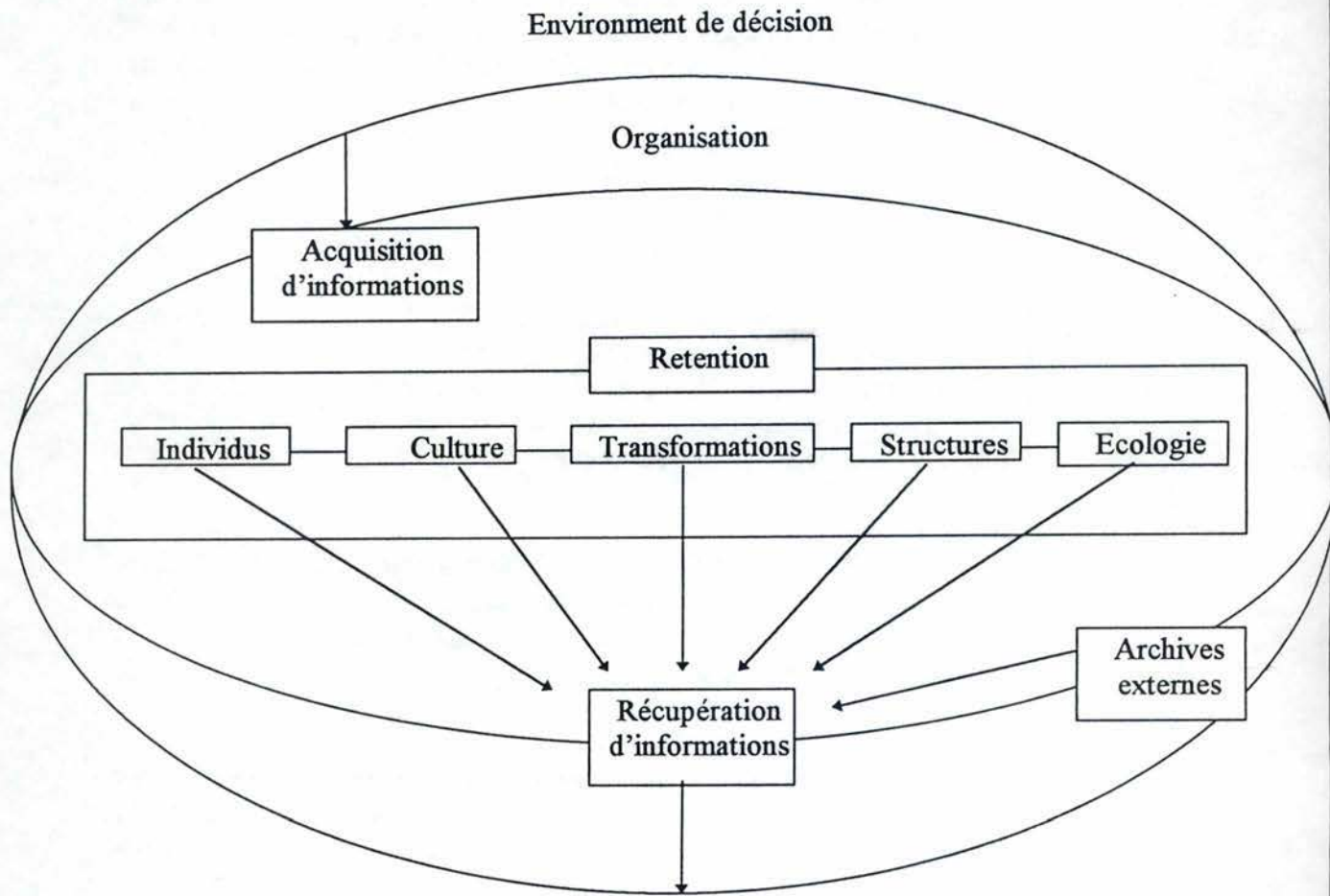


Figure 1.2. : Le modèle de mémoire organisationnelle selon Walsh et Ungson - Adapté de Walsh et Ungson (1991)

### La récupération

Toute information mémorisée est, par essence, susceptible d'être récupérée à un moment donné. Walsh et Ungson présentent la récupération selon un continuum allant de la récupération automatique (sans effort) à la récupération contrôlée (avec efforts). Ils rangent les transformations, les structures, la culture et l'écologie dans les récupérations du premier type, tandis que les individus et la culture se voient liés à la récupération de second type.

Certaines critiques ont été énoncées par rapport à ce modèle. La principale d'entre-elles concerne le manque de considération qui est portée aux systèmes d'information



(Stein et Zwass (1995)). En effet, le rôle accordé par Walsh et Ungson (1991) à ces systèmes se limite au stockage d'enregistrements et de fichiers en tant qu'aide-mémoire. Comme autre critique, nous citerons encore Bannon et Kuutti (1996) qui jugent que le modèle essaye d'englober virtuellement tout, ce qui rend finalement difficile la distinction entre ce qui est mémoire organisationnelle et ce qui ne l'est pas.

### 1.7.2. Modèle de Stein et Zwass

Dans leur modèle, Stein et Zwass décrivent un cadre de travail conceptuel pour les systèmes d'informations supportant de la mémoire organisationnelle (SISMO). Ils ne décrivent pas un SISMO particulier, mais plutôt un ensemble de caractéristiques présentes, de manière générale, dans les SISMOs. La création de ce modèle est justifiée par la force de cohésion qui existe entre la mémoire organisationnelle et les systèmes d'information.

Le modèle de Stein et Zwass (Figure I.3.) est un modèle en deux couches construit sur base de quatre fonctions d'efficacité organisationnelle présentées par Quinn et Rohrbaugh (1983).

Sous-système Intégratif	Sous-système Adaptatif	Sous-système de Réalisation des buts	Sous-système de Maintenance des procédures de travail
Fonctions Mnémoniques (acquisition, rétention, maintenance, recherche et récupération de connaissances)			

Figure I.3. : Modèle du cadre de travail des SISMOs - Adapté de Stein et Zwass (1995)

La couche inférieure du modèle reprend l'ensemble des fonctions mnémoniques qui permettent de gérer la mémoire (acquisition, rétention, maintenance, recherche et récupération de connaissance).

La couche supérieure reprend un ensemble de sous-systèmes qui supportent directement les quatre fonctions d'efficacité présentées dans le modèle de Quinn et Rohrbaugh, à savoir :

- la fonction *Intégrative* gérant la coordination de l'organisation et la circulation des informations au sein de l'organisation.

- la fonction *Adaptative* offrant les moyens à l'organisation de s'adapter aux changements de son environnement.
- la fonction *Réalisation de but* permettant à l'organisation de définir ses buts et d'évaluer leur degré de réalisation.
- la fonction *Maintenance des procédures de travail* permettant de maintenir la cohésion et le moral de la force de travail.

Les 4 sous-systèmes de la couche supérieure sont :

#### **Le sous-système Intégratif :**

Le sous-système Intégratif doit assurer que les connaissances sont bien partagées et disponibles dans le temps et dans l'espace. L'intégration dans l'espace comprend l'utilisation de bases de données et de bases de connaissances partagées ainsi que la distribution du travail à travers les réseaux de communication. L'intégration dans le temps comprend des bases de données d'événements, de problèmes et de modèles indexées dans le temps et enrichies par le multimédia.

#### **Le sous-système Adaptatif :**

Le sous-système Adaptatif doit pouvoir définir la frontière du domaine des activités pour reconnaître, capturer, organiser et distribuer la connaissance relative à l'environnement aux acteurs appropriés de l'organisation. Ce sous-système doit mémoriser des informations chronologiques sur le marché, gérer des bases de données sur les préférences des utilisateurs et disposer de liens avec des sources d'informations externes.

#### **Le sous-système de Réalisation de but :**

Le sous-système Réalisation de but doit permettre d'aider à identifier les buts dans le contexte du passé de l'organisation, à stocker les buts de l'organisation, à formuler des stratégies pour atteindre les buts, à évaluer le progrès en direction de la réalisation des buts, à suggérer des alternatives basées sur des évaluations, à mettre à jour les buts sur base de nouvelles informations et à stocker des histoires annotées. Ce sous-système doit inclure des experts planifiant la connaissance, des modèles d'évaluation, des données sur les performances de la société, des prévisions passées et actuelles avec leurs variations



par rapport aux attentes, les engagements passés et présents triés dans le temps ainsi que les circonstances dans lesquelles ces engagements ont été pris.

**Le sous-système de Maintenance des procédures de travail :**

Le sous-système de Maintenance des procédures de travail doit agir au niveau des individus et au niveau de l'organisation. Il doit mémoriser l'historique de travail des individus en mettant en évidence la description des projets auxquels ils ont participé, leurs capacités, leurs niveaux techniques et leurs aspirations. La formation des individus doit aussi être soutenue par ce sous-système ainsi que les protocoles de l'organisation et leurs valeurs implicites. Ce sous-système doit inclure des descriptions riches et partielles de cas, de projets, de procédures, de contributions du personnel et d'indicateurs sur le climat et la culture de l'organisation, le tout pouvant être supporté par du multimédia.

***1.7.3. Modèle de Morrison et Weiser***

Les deux modèles proposés, d'une part, par Walsh et Ungson et, d'autre part, par Stein et Zwass concernent chacun une partie de la mémoire organisationnelle. Le premier modèle se concentre sur les "coffrets" d'information présents dans les organisations, tandis que le second fait intervenir les systèmes d'information comme composante essentielle de la mémoire organisationnelle. Face à ces deux modèles, Morrison et Weiser (1996) proposent un modèle général (voir Figure I.4.) qui assure, en quelque sorte, l'unification des deux autres modèles.



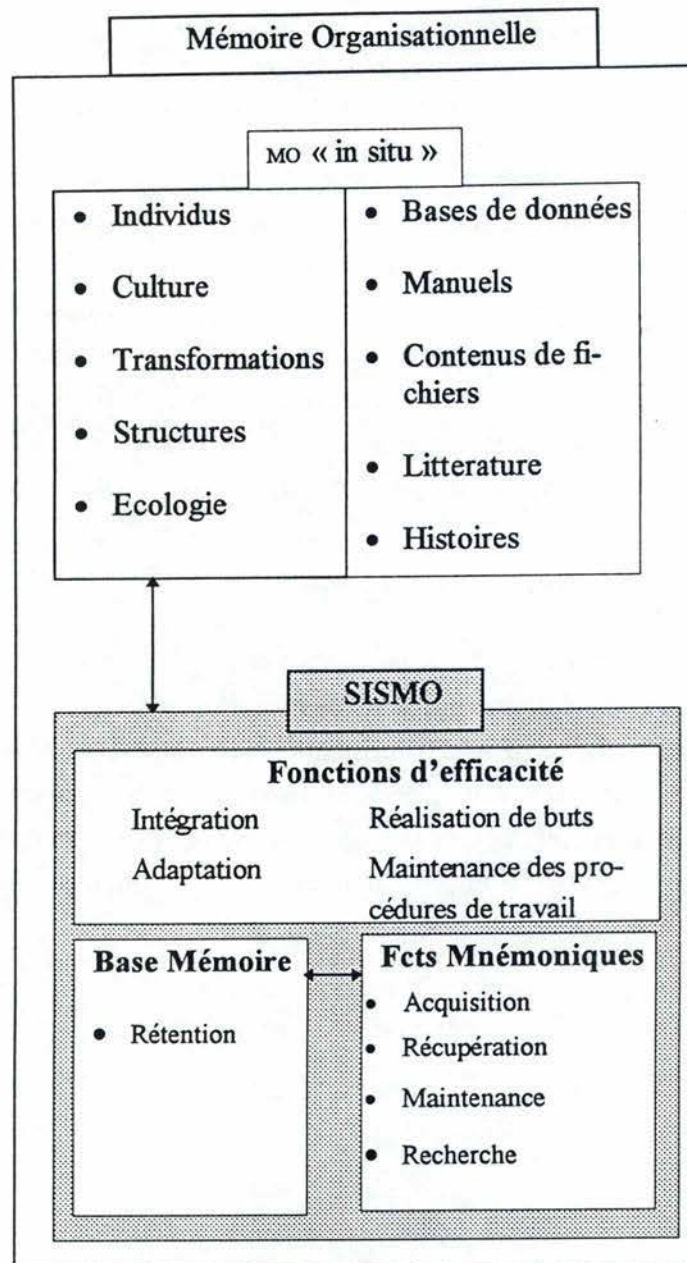


Figure I.4. : Structure de la mémoire organisationnelle généralisée - Adapté de Morrison et Weiser (1996).

Dans ce modèle, la mémoire organisationnelle est composée de deux parties : "MO *in situ*" et "SISMO".

La mémoire *in-situ* peut se définir comme étant la mémoire qui existe dans une organisation et qui n'est pas supportée par un SISMO. Elle peut être présente dans différents

référentiels (les individus, la culture, les bases de données, etc...). Cette partie de la mémoire organisationnelle peut interagir avec un ou plusieurs SISMOs représentés sur base du modèle de Stein et Zwass (1995). L'interaction s'opère lorsque les SISMOs ont besoin d'informations présentes dans l'organisation et qui peuvent servir de support à l'activité qu'ils mémorisent. Par exemple, dans le cas d'un SISMO lié aux prises de décisions, la culture de l'organisation peut intervenir dans le processus de prise de décision.

Les SISMOs sont modélisés sous la forme de deux couches (voir modèle de Stein et Zwass au point I.7.2.). Le niveau supérieur reprend les quatre fonctions du modèle de Stein et Zwass. Au niveau de la couche inférieure, Morrison et Weiser (1996) distinguent les fonctions mnémoniques de la base mémoire. La base mémoire d'un SISMO contient les données qui sont liées aux activités qu'il supporte. Si un SISMO s'occupe de mémoriser des prises de décision, la base mémoire contiendra les données qui sont nécessaires pour, par exemple, retracer les différentes étapes de la prise de décision. Dans ces données se retrouveront des liens vers la mémoire *in-situ* (e.g. un article de presse stocké dans une base de données). Inversement, la mémoire *in-situ* peut aussi être influencée par un SISMO. Par exemple, si un individu participe à une prise de décision supportée par un SISMO, il aura modifié sa mémoire, et indirectement la mémoire *in-situ*.

Morrison (1995) et Morrison et Weiser (1996) détaillent les fonctions mnémoniques, également présentes dans le modèle de Stein et Zwass :

**L'acquisition** : fonction dont la tâche est de placer des informations dans la base mémoire. Nous pouvons distinguer trois types d'informations :

- Les **archives** : informations placées volontairement dans la base mémoire par des agents humains soit au démarrage du système, soit à la terminaison d'une tâche.
- Les **réponses** : certaines personnes, ayant besoin d'informations à propos d'un sujet, posent des questions à des experts humains. Ceux-ci y répondent et les réponses sont stockées dans la mémoire organisationnelle et rendues accessibles à tous les utilisateurs du système.
- Les **insertions automatiques** : les informations sont automatiquement insérées dans le système grâce à certains des processus (e.g. systèmes d'aide à la négociation) qui jouent le rôle de fournisseurs artificiels d'information.

**La recherche et la récupération** : les informations ayant été mémorisées doivent évidemment pouvoir être récupérées. Selon Huber (1991), cette récupération peut être **active** (consciente, désirée) ou **passive** (récupération sans efforts ou fortuite).



**La recherche active :** Différentes techniques permettent de récupérer des données de manière active (Morrison (1995) et Morrison et Weiser (1996)):

- **Les requêtes :** fonctions qui permettent de récupérer, à l'aide de clés de recherche, des informations dans la mémoire organisationnelle. Les clés de recherche sont différentes suivant qu'il s'agisse de récupérer des documents, des données liées à un processus (e.g. prise de décision, négociation) ou des données de connaissances et d'expertise. Une requête sur les documents peut, par exemple, comprendre le nom de l'auteur d'un document, la date de création de ce document; une requête sur les processus peut porter sur les acteurs de ce processus et une requête sur les connaissances peut faire référence aux sources humaines de ces connaissances.
- **Le filtrage :** fonction qui permet de faciliter les recherches des utilisateurs. L'abstraction est un moyen de filtrer l'information. L'utilisateur consulte des abstracts de documents plutôt que les documents en entier (cf. bibliothèques). La présentation des étapes clés d'un processus constitue aussi une forme d'abstraction. Le filtrage par date constitue une autre approche pour filtrer les informations dans lesquelles seuls les éléments récents sont présentés. Enfin, le filtrage du contexte permet de restreindre la présentation d'un processus suivant un certain angle de vue (e.g. fonction *view* dans les SGBDs).
- **La navigation :** requête stockée qui crée des liens entre des données de base données avec des données d'autres types. Un document (une donnée de type non BD) peut, par exemple, posséder un lien vers une donnée BD.
- **L'exploration guidée :** méthode utilisée dans les pages Web sur Internet. Les documents possèdent des liens vers d'autres pages et guident, de la sorte, l'utilisateur dans son exploration. D'autre part, des moteurs de recherche sont aussi mis à la disposition du *surfer* afin de faciliter ses explorations.
- **Demande à un expert :** consiste à interroger directement (via courrier électronique par exemple) un expert humain dans le domaine qui pose un problème à l'utilisateur.

**La recherche passive :** cette recherche peut être effectuée soit en scannant l'environnement à la recherche d'informations, soit en remarquant, par hasard, des faits internes ou



externes à l'entreprise (Huber (1991)). Morrison remarque que les principales sources de recherche passive sont, d'une part, les contacts informels internes ou externes à l'organisation (e.g. les conversations autour d'une tasse de café ou sur un cours de golf), et d'autre part, les communications internes formelles (circulaires, mémorandum, listes de distribution de messages électroniques, etc...).

**La maintenance** : fonction qui assure la gestion de la mémoire de base dans le but de

- respecter la confidentialité des données
- assurer le contrôle d'accès et l'intégrité des données
- permettre l'oubli de certaines informations obsolètes ou inutiles (voir dangers de la mémoire organisationnelle)

## ***1.8. Les systèmes supportant la mémoire organisationnelle***

---

Dans cette section, nous nous intéressons à la relation entre la mémoire organisationnelle et les systèmes d'information. Dans un premier temps, nous nous occupons à montrer pourquoi il est intéressant, sinon nécessaire, d'utiliser et développer des systèmes d'information qui supportent la mémoire organisationnelle. Nous analysons le champ d'action potentiel de ces systèmes car il s'agit aussi d'étudier si ces systèmes sont ou ne sont pas capables de supporter l'ensemble de la mémoire organisationnelle. Ensuite, nous exposons les exigences de ces systèmes d'information, ainsi que les difficultés qu'il faut contourner pour les implémenter.

### ***1.8.1. Pourquoi développer les SISMOs ?***

Stein et Zwass (1995) relèvent plusieurs points qui plaident en faveur du développement des SISMOs. D'abord, ils prétendent que la complexité croissante de l'environnement et l'augmentation des rotations de personnel rendent nécessaires de tels systèmes. Ils vont même plus loin en affirmant que la mémoire organisationnelle aujourd'hui et dans le futur reposera sur des systèmes informatiques structurés qui auront été créés dans ce but : « *In our view, organizational memory today and in the future will rely on expressly structured information systems support* » (p 90). Toujours selon Stein et Zwass, l'élargissement du répertoire des SISMOs aide les acteurs humains à faire face à une surcharge d'informations et assume leur rôle en tant que « processeurs d'information ». Finalement, ils prétendent que les technologies de l'information, lors-

qu'elles sont utilisées proprement, mènent les activités de l'organisation à des niveaux d'efficacité et d'apprentissage plus élevés. La hausse du niveau d'efficacité provient entre autres de la chute des prix des technologies informatiques.

Ces différentes idées concordent avec la prévision de Huber (1990) qui considère que le développement des technologies de l'information mène au développement et à l'utilisation de bases de données et de systèmes experts en tant que composants de la mémoire organisationnelle.

Balasubramanian (1995) est favorable au développement des SISMOs car ils influencent directement et indirectement l'apprentissage. Les SISMOs influencent indirectement l'apprentissage car ils peuvent modifier les facteurs contextuels, comme la structure ou l'environnement. L'introduction d'un système informatique dans une organisation peut aplanir sa structure et promouvoir une plus grande diffusion de l'information. Balasubramanian (1995) prétend aussi que les SISMOs peuvent directement répondre aux besoins des processus d'apprentissage exposés par Huber (1991) (voir I.5.3.).

### ***1.8.2. Comment utiliser les SISMOs ?***

Il existe déjà des systèmes informatiques qui supportent de la mémoire organisationnelle (voir point I.10.). Ils se différencient en fonction du domaine qu'ils visent. Certains systèmes tentent de capturer une certaine partie des connaissances d'une entreprise ou d'une communauté scientifique, tandis que d'autres se limitent à gérer la mémoire organisationnelle liée à une tâche particulière. Ackerman et Mandel (1995) prétendent que les mémoires basées sur la tâche seront hautement importantes pour les organisations car les SISMOs ont besoin d'un contexte suffisant pour récupérer l'information mémorisée. L'indexation, la classification de l'information pertinente est très difficile, voire impossible, dans le cas de problèmes compliqués. Ils ajoutent aussi que les organisations ont des ressources et attentions limitées ce qui rend la mémoire organisationnelle plus utile quand elle est attachée aux besoins à court terme.

### ***1.8.3. Les qualités d'un SISMO***

Nous présentons maintenant les différentes qualités qu'un SISMO doit pouvoir posséder (voir Tableau I.2.). Ces qualités sont relevées par Watson (1996) et nous les résumons ci-après.



Partageable	Peut être accédé par plusieurs personnes en même temps.
Transportable	Offre une communication aisée des données.
Sûr	Est protégé de la destruction et de l'usage non autorisé.
Précis	Offre des données précises à propos desquelles les utilisateurs peuvent avoir confiance.
A jour	Offre des données actuelles, <i>on-line</i> .
Pertinent	Approprié au domaine considéré

Tableau 1.2.: Les attributs attendus d'un SISMO - adapté de Watson (1996)

### Partageable

Tous les membres de l'organisation doivent pouvoir accéder à la mémoire organisationnelle, lorsqu'ils ont besoin d'informations pour accomplir leur tâche. Des situations dans lesquelles plusieurs personnes désirent consulter les mêmes données simultanément doivent pouvoir être supportées par un SISMO.

Les organisations utilisent plusieurs approches pour partager les informations. Le succès de ces différentes méthodes dépend principalement de la volatilité des données. Des données qui sont souvent modifiées, sont plus difficiles à gérer (e.g. contrôle d'accès concurrents) que les données statiques.

### Transportable

Les données doivent pouvoir être transportées depuis leur endroit de stockage vers les utilisateurs qui en ont besoin. Au cours de l'histoire, les différents moyens de communication ont évolués : le pigeon voyageur a été remplacé par l'ordinateur qui permet le transport de données textuelles, visuelles et vocales.

De nos jours, la *transportabilité* des données ne se limite pas au transfert d'informations vers le bureau d'un preneur de décision, mais permet aussi, par exemple, aux clients d'accéder à des données directement depuis leur propre bureau. Les preneurs de décision doivent pouvoir avoir accès à de l'information où et quand ils le désirent, malheureusement, la plupart des organisations sont encore loin de cet objectif.

## **Sûr**

Les données constituent des ressources de haute valeur pour les entreprises. Etant donné l'importance de la mémoire pour une organisation, des précautions contre le vol, le sabotage ou la perte doivent être prises pour ne pas, un jour, souffrir d'amnésie (e.g. effectuer des copies, séparées géographiquement). Un moyen de protéger le système est de surveiller les accès physique (portes fermées à clés) et électronique (par l'introduction d'un mot de passe).

## **Précis**

La qualité des informations est importante. Il est certain que des organisations qui disposent de données précises seront plus efficaces que celles qui disposent de données vagues. Les compagnies aériennes constituent un bon exemple. Nous pouvons aisément imaginer quels problèmes pourraient survenir si, par exemple, une compagnie ne connaissait pas précisément le nombre de places disponibles dans l'un de ses avions. La pollution des données est un problème critique pour les organisations. En fait, la plupart des processus d'une organisation sont directement liés aux données et il faut donc que celles-ci soient correctes sous peine d'aboutir à des mauvaises prises de décision, des retards dans les livraisons, des erreurs de factures, etc... L'image de marque d'une entreprise dépend de la qualité de ses informations. Il faut donc que les organisations soient prudentes au moment où elles capturent des informations, c'est-à-dire le moment où la précision des données est la plus vulnérable.

## **A jour**

La valeur d'une donnée est souvent déterminée par son âge et peut rapidement dépérir avec le temps. Les preneurs de décision ont souvent besoin d'accéder à des données en direct (e.g. La Bourse), pour pouvoir réagir le plus vite possible. Par contre, des données anciennes sont aussi utiles car elles permettent d'analyser les tendances à long terme.

## **Pertinent**

L'objectif de la mémoire organisationnelle est de maintenir des données qui sont pertinentes pour ses activités (cf. définition de la mémoire organisationnelle). La pertinence des données n'est pas facile à évaluer car il faut aussi bien considérer les données qui supportent les activités présentes, mais aussi celles qui pourraient soutenir les activités futures de l'organisation. Un filtrage au niveau de la récupération des informations permet d'offrir un système qui ne conserve que les données pertinentes par rapport à une activité particulière.



Nous pouvons remarquer que les qualités présentées par Watson (1996) se retrouvent déjà dans les systèmes d'information classiques. Il s'agit donc d'intégrer dans les SISMOs les technologies existantes telles que celles qui apparaissent dans les SGBDs ou les réseaux.

#### **1.8.4. Les problèmes des SISMOs**

Le chemin des concepteurs de SISMOs est parsemé d'embûches et il est nécessaire de les surmonter pour que ces systèmes soient efficaces. Watson (1996) relève les difficultés rencontrées par les SISMOs (voir Tableau I.3.). Nous résumons les points développés par Watson (1996).

Redondance	La même donnée est stockée à différents endroits.
Le contrôle des données	Les données sont mal gérées.
Interface	Difficultés pour accéder aux données.
Retards	Longs temps de réponse pour les requêtes sur les données.
Manque de réalisme	Les mémoires organisationnelles ne reflètent pas la complexité du monde réel.
Manque d'intégration des données	Les données sont distribuées à travers des systèmes différents.

*Tableau I.3.: Les problèmes attendus d'un SISMO - adapté de Watson (1996)*

##### **Redondance**

Généralement, les mémoires organisationnelles ont grandi au petit bonheur la chance. De ce fait, une même donnée est souvent stockée dans différentes mémoires. L'exemple classique est celui des données (nom, adresse, tél., ...) d'un client qui sont copiées dans différents systèmes, ce qui pose des problèmes lors d'une mise à jour.

La redondance apporte de la confusion dans les données et est directement liée à l'intégrité de ces données.

### **Contrôle des données**

Le contrôle des données est lié au problème de la redondance. La plupart du temps, les organisations consacrent plus d'attention aux ressources telles que l'argent et le personnel et négligent quelque peu l'importance des données. Seuls les départements de type MIS (*Management Information Systems*) s'occupent de capturer des données, mais cette récolte se limite souvent aux données nécessaires pour les systèmes informatiques. Si un département de l'organisation n'est pas informatisé, aucune gestion de ses données n'est opérée.

### **L'interface**

L'interface constitue un problème clé des SISMOs. Le découragement des utilisateurs potentiels des SISMOs a souvent trouvé son origine dans une interface désagréable. Les utilisateurs sont frustrés parce qu'ils reçoivent des messages d'erreur incompréhensibles ou que leurs requêtes sont refusées par le système.

### **Retards**

Au cours de ces dernières années, la vitesse à laquelle se déroulent les affaires d'une entreprise a augmenté. Les dirigeants doivent se décider plus vite et exigent, dès lors, que les systèmes informatiques puissent répondre rapidement à leurs demandes.

### **Manque de réalisme**

Les mémoires organisationnelles doivent refléter la réalité et la complexité du monde réel afin de pouvoir répondre de manière rapide et efficace aux différentes requêtes. Elles doivent pouvoir répondre aux attentes des utilisateurs du système. Dans le cas d'une prise de décision, par exemple, un décideur doit pouvoir interroger la mémoire organisationnelle aussi bien à propos de rapports coutumiers que de rapports *ad-hoc*.

### **Manque d'intégration des données**

La plupart des organisations souffrent d'un manque d'intégration des données. Non seulement, les informations sont dispersées sur des supports de mémorisation différents (e.g. disque dur, CD-ROM), mais manquent aussi d'intégration au niveau de leur format.

Watson (1996) prévoit une continuation de ce manque d'intégration. D'abord, les systèmes informatiques qui ont été développés dans le passé ne disposaient pas des moyens technologiques actuels leur permettant d'intégrer les données. Etant donné que la plupart de ces systèmes sont encore utilisés, l'intégration des données est freinée. De plus, l'inté-



gration est une politique à long terme. Il s'agit de créer de nouveaux systèmes et de réécrire les anciens de manière à limiter les coûts et les risques de désorganisation.

## ***1.9. Exemples de SISMOs***

---

Différents systèmes ont déjà été élaborés pour supporter, de façons diverses, de la mémoire organisationnelle. Nous reprenons ci-après quelques exemples de systèmes supportant de la mémoire organisationnelle. Nous présentons ces exemples sur base d'un classement établi par Morrison (1995), à savoir : les systèmes d'intégration de documents, les systèmes de support à l'activité humaine et les systèmes de capture et d'assimilation des connaissances. Nous terminons cette section, par la présentation de SISMOs émergents.

### ***1.9.1. Les systèmes d'intégration de documents***

Les systèmes d'intégration de documents se séparent en deux classes :

- **Les systèmes de gestion de documents** : Systèmes qui scannent des documents pour en obtenir des images. Les documents sont organisés par un système d'index interrogé par des mots-clés. Il s'agit du système de plus en plus présent dans les bibliothèques, où l'on ne consulte plus les articles depuis des magazines, mais depuis des CD-ROM. Dans les systèmes de gestion de documents se trouvent aussi les systèmes qui stockent des documents informatiques (Word, Excel, ...) identifiés par leur nom.
- **Les systèmes Hypertexte/Hypermédia** : Dans ces systèmes, les documents sont composés de réseaux de noeuds et sont liés les uns aux autres d'une manière logique et physique. L'hypermédia jongle avec des éléments tels que le son, l'image, ce que ne permet pas l'hypertexte.

### ***1.9.2. Les systèmes de support à l'activité humaine***

Nous nous intéressons à deux types de systèmes qui supportent des activités humaines:

- **les systèmes d'aide à la décision** : systèmes d'information interactifs fournissant des informations, des outils ou des modèles pour aider les dirigeants ou les professionnels à prendre des décisions (Alter (1992)). Morrison (1995) prétend que les gestionnaires utilisent depuis toujours des informations, des outils et des modèles pour prendre leurs décisions. Par contre peu d'entre-eux utilisent des systèmes supportés par l'informatique et cela est peut-être dû au fait que les données des systèmes d'aide à la décision sont difficiles à capturer, stocker et récupérer.
- **les systèmes d'aide à la coopération** : cette catégorie de systèmes, également appelée *groupware*, supporte le travail de groupe et intègre (Watson (1996)) :
  - des applications telles que le courrier électronique, les vidéoconférences, des logiciels de gestion d'agenda et d'horaires;
  - un logiciel de workflow qui automatise le routage des processus et des formulaires de gestion à travers un réseau;
  - un logiciel de gestion de documents qui permet de créer et de réviser des documents partagés dans un réseau;
  - un logiciel de support au travail de groupe pour faciliter les réunions électroniques.

Comme logiciel groupware, nous présentons le système **GroupSystem 5.0** (Martz (1991), Watson (1996)). Ce système peut supporter des réunions dans lesquelles les participants sont séparés dans le temps et dans l'espace. *Groupsystem* fournit des outils de support aux différentes étapes présentes dans une réunion : la génération, l'organisation, l'évaluation et l'établissement des priorités d'idées. Les outils peuvent être regroupés selon quatre catégories : la génération d'idée, l'organisation d'idée, l'évaluation d'idée et l'exploration du problème (voir Figure I.5.).



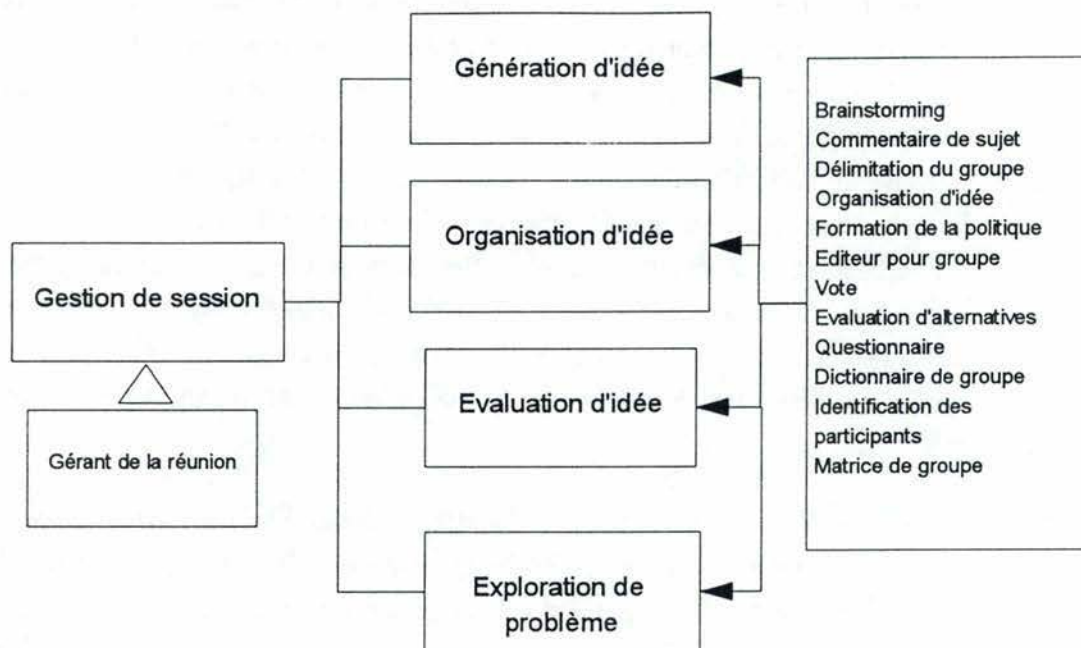


Figure 1.5. : Fonctions offertes par GroupSystem 5.0 - Adapté de Martz (1991)

Ces systèmes sont de plus en plus répandus, cependant ils manquent encore d'efficacité pour capturer les processus de groupe ainsi que le contexte (Morrison (1995)). Nous présentons ici, trois systèmes qui essaient de capturer le contexte et les informations à propos des processus de groupe :

- **gIBIS** (Conklin et Begeman (1989)) : *graphic Issue Based Information System*. Ce système hypertexte a été conçu pour faciliter et capturer des discussions entre différents acteurs impliqués dans un problème de conception. Il implémente une méthode spécifique *IBIS* qui a été développée pour être utilisée dans le cas de problèmes complexes. Le modèle *IBIS* se concentre sur une articulation d'éléments critiques (*Issue*) dans un problème de conception. Chaque *Issue* peut avoir différentes *Positions* supportées ou attaquées par des *Arguments*. Le but du système est de permettre à tous les acteurs de comprendre les points de vues émis par les autres participants.

- **Sibyl** (Lee (1990)) : ce système supporte la résolution des problèmes de groupe. Son but principal est de représenter explicitement la connaissance relative à une décision afin que tous les preneurs de décision comprennent le problème de la même façon. Le processus de prise de décision est structuré par une formalisation des buts, des alternatives et des arguments. Les participants peuvent émettre des arguments et des contre-arguments. Toutes les interventions sont mémorisées ce qui offre le moyen aux utilisateurs de retrouver des informations relatives à un problème passé ou présent. Sibyl représente graphiquement le problème de décision, les éventuelles alternatives, les problèmes liés à ces alternatives et les revendications par rapport à l'un de ces éléments.
- **CGSE** (Sandoe et al. (1991)) : *Claremont GDSS Environment*. Ce système cherche à capturer des processus de groupe dans des processus de réunions génériques. Ce système comprend des outils pour le travail individuel et collectif, accède à des informations relatives à des réunions précédentes, à des fichiers extérieurs et à un système de messagerie. Il offre aussi des fonctions automatiques liées aux actions spécifiques présentes dans une réunion : retrait, modification, présentation d'une motion, etc... . CGSE utilise un système de récupération de type hypertexte autorisant les utilisateurs à *voyager* parmi les actions d'une réunion présente ou passée. Ce système, offrant à tous les membres d'un groupe une vue globale sur le processus, réduit toute ambiguïté entre les participants d'une réunion.

### **1.9.3. Les systèmes de capture et d'assimilation de connaissances**

La connaissance est généralement capturée à l'aide de systèmes experts basés sur des règles. La plupart du temps, ces systèmes se limitent à répondre à des questions simples et ne peuvent résoudre que des problèmes structurés. Ces systèmes ne peuvent pas apprendre, analyser un problème sous un autre angle ou représenter leur raisonnement (Winston (1984)).

### **1.9.4. Les SISMOs qui surgissent**

Au cours des trois derniers points, nous avons présenté différents systèmes qui supportent des éléments de mémoire organisationnelle : certains gèrent des documents, des informations liées aux prises de décision ou de la connaissance, tandis que d'autres favori-



sent la coopération au sein d'un groupe. En fait, sur base des modèles que nous avons présentés précédemment, nous pouvons affirmer que les SISMOs devraient pouvoir reprendre des caractéristiques de ces différents systèmes, car ainsi, les utilisateurs pourraient intégrer des bases de données, des documents, des connaissances sur des processus, des éléments de culture et de littérature afin de résoudre des problèmes, prendre des décisions et accomplir des tâches diverses. Nous présentons, ci-après, différents logiciels qui permettent de développer des outils de support à la mémoire organisationnelle.

### **Answer Garden Substrate**

*Answer Garden Substrate* a été créé pour permettre la création de nouvelles applications supportant la mémoire organisationnelle ou pour étendre celles qui existent déjà. Ce système est formé des composants suivants :

- Un **service d'interface** à l'utilisateur constitué d'un ensemble d'objets de présentation permettant de supporter l'interface
- Un **service de fichiers** qui offre des liens vers les référentiels des données
- Des **moteurs de récupération d'information** qui proposent plusieurs stratégies de recherche et de récupération des données
- Un **service de communication** qui supporte le courrier électronique et la messagerie synchrone.

Deux principales applications ont été développées à l'aide de *Answer Garden Substrate* : *Answer Garden* (Ackerman (1994)) et *The Assist* (Ackerman et Mandel (1995)).

*Answer Garden* est un système réseau hypermédia qui marie le monde des bases de données et celui de la communication. Lorsque des utilisateurs se posent une question à propos d'un sujet, ils peuvent chercher la réponse dans un système qui conserve les questions (et leurs réponses) qui ont été le plus souvent posées. Si la question de l'utilisateur n'est pas reprise dans le système, elle est immédiatement envoyée, par courrier électronique, à un expert humain, dont la réponse sera enregistrée dans le système. Ackerman a reconnu que son système souffrait d'un manque de représentation du contexte. De manière à pallier cette lacune, il préconise une utilisation locale et à court terme du système car l'interprétation est plus facile lorsqu'elle s'effectue entre des personnes qui partagent déjà la même culture.



*The Assist* est un système qui supporte de la mémoire organisationnelle liée à une tâche pour toute une communauté scientifique. Il permet aux utilisateurs de systèmes complexes de recevoir de l'aide d'autres utilisateurs de ces systèmes. *The Assist* sert de mémoire pour l'organisation, mais aussi d'anti-mémoire car il peut oublier les détails superflus qui ne sont pas liés à la tâche. Avec *The Assist*, Ackerman a revu ses prétentions quelque peu à la baisse puisque, pour des raisons essentiellement liées à la gestion du contexte et des ressources, il limite l'utilisation de son système à une tâche particulière. Il ne s'agit plus de concevoir un système qui supporte toute la mémoire organisationnelle, mais seulement la mémoire d'une tâche spécifique (*Memory in the small*).

### **Lotus Notes**

Lotus Notes (Hutchison (1995)) est un système logiciel pour PC's et stations de travail mis en réseau et qui permet le travail en groupe (*workgroup*). Un *workgroup Notes* comprend un groupe de personnes qui partagent des informations et travaillent ensemble régulièrement. Les membres du groupe peuvent être à un même endroit ou distribués géographiquement. L'application Notes consiste en un serveur Notes et des stations de travail Notes. Le serveur fournit des services de bases de données partagées et de routage de mails vers les stations de travail. L'environnement Notes fournit aussi des services tels que le courrier électronique et les valves électroniques.

Lotus Notes fournit une plate-forme pour des applications de groupe qui collecte, organise et distribue de l'information semi-structurée, s'occupe du routage des documents ainsi que du workflow. L'architecture Lotus Notes est une base de données distribuée qui maintient des copies des données sur des sites géographiquement distribués. Dès que certains membres d'un groupe modifient des données, celles-ci sont mises à jour sur les autres sites. Lotus Notes est particulièrement populaire et de nombreuses applications sont développées avec des données dont le format est compatible avec Lotus Notes, ce qui favorise l'intégration de différentes plates-formes de documents et de bases de données.

### **Worm Community Database**

Le *Worm Community project* (Schatz (1992)) consiste à créer un système électronique pour une communauté scientifique qui encode la connaissance de cette communauté et fournit un environnement de support à la manipulation de cette connaissance. La base de données contient des informations intéressantes pour la communauté : des données archivées (descriptions de gènes, de cartes génétiques, ...), de la littérature (bibliographies, résumés, des images, des articles), des informations informelles (circulaires, résumés de



papiers de conférence), des données non-publiées (résultats expérimentaux intermédiaires) et la connaissance de la communauté développée par l'entremise du système. La manipulation du système permet de parcourir la connaissance disponible mémorisée dans ce système, ainsi que de créer des liens d'interconnexion entre ses différents éléments.

### **Team Memory**

*Team memory* (Morrison (1993)) est un système qui assure la coopération au sein d'un groupe de travail. Le système fournit un environnement de projets qui aide les membres à organiser et à suivre à la trace les données partagées. Le système offre, par un jeu d'icônes, les moyens d'accéder aux données des différents membres de l'équipe, ainsi qu'aux éléments contextuels (liste des participants, buts du projet, tâches à faire, ...). Il est aussi possible de rechercher d'anciens projets sur base de leurs participants, de mots-clés, ou de leur titre.

## ***1.10. Critiques du concept et des modèles de mémoire organisationnelle***

---

Jusqu'à présent, nous nous sommes concentrés sur les différentes idées qui ont été développées en matière de mémoire organisationnelle. Il nous faut cependant poser un regard critique sur ce concept. Dans cette section, nous critiquons la mémoire organisationnelle selon deux angles de vue. D'abord, nous déterminons si le concept de mémoire organisationnelle est vraiment neuf, ou s'il s'agit simplement d'un nouveau nom associé à un concept déjà existant. Ensuite, nous soulevons certaines critiques par rapport aux différents modèles de mémoire organisationnelle que nous avons présentés.

Il convient de se demander si les SISMOs constituent quelque chose de neuf par rapport aux systèmes d'information traditionnels. Nous allons essayer de comprendre s'il s'agit d'un outil supplémentaire destiné à un certain type d'applications ou bien d'une nouvelle façon de construire un système d'information pour une organisation. Nous ne voulons pas établir un relevé exhaustif des types de systèmes d'information existants, mais nous relevons toutefois trois catégories de systèmes d'information :

- **Les systèmes d'information opérationnels** qui ont pour but de servir de support à des processus de gestion d'une organisation. Les données et les fonctions qu'ils comprennent ont été établis de manière à faciliter, voire automatiser, un ou plusieurs processus de gestion. Les informations présentes dans ce genre de systèmes sont exclusivement liées à la tâche qu'ils exécutent.



- **Les systèmes d'information pour dirigeants** (*Executive Information Systems -EISs*) qui sont des systèmes de recueil d'informations de toute nature, de présentation sous forme de tableaux de bord, de manipulation aisée et d'exploration fine de ces informations (Courbon (1993)). Les EISs ont été conçus de manière à synthétiser les différentes données présentes dans une organisation. Généralement, la synthèse se présente sous la forme d'un rapport.
- **Les systèmes d'information d'aide à la décision** (SIADs) qui sont des systèmes d'information destinés à supporter les prises de décision dans une organisation. Les SIADs (*Decision Support Systems*) peuvent être spécialisés en différentes catégories : les GDSSs (prises de décision de groupe), les NSSs (*Negotiation Support Systems*), ... .

Les systèmes d'information opérationnels travaillent au niveau le plus bas de l'organisation. Par contre les EISs et les SIADs répondent à des besoins des dirigeants, des preneurs de décision. Ces systèmes sont supportés par des référentiels de données appelés infocentres (Courbon (1993)) et, plus récemment, *datawarehouse* (de Linares (1996)).

L'infocentre est un concept né il y a une quinzaine d'années pour permettre aux membres de l'organisation d'accéder eux-mêmes aux données informatiques. Les départements informatiques, étant trop chargés que pour systématiquement répondre aux requêtes externes au département, ont constitué un système contenant un condensé du contenu des différentes bases de données opérationnelles (Courbon (1993)). L'interface était spécialement étudiée pour permettre un accès aisé à des individus non familiarisés avec l'usage des ordinateurs.

Actuellement, les infocentres sont de plus en plus remplacés par les *datawarehouses* (de Linares (1996)). Ce terme désigne une nouvelle génération d'infocentres plus performants. L'apport du *datawarehouse* par rapport à l'infocentre se situe au niveau de la qualité d'intégration des données et de leur analyse (de Linares (1996)). Les quantités d'informations stockées grandissent en permanence. Dans une organisation, chaque département - depuis la comptabilité jusqu'au marketing - possède ses informations opérationnelles. Celles-ci sont rassemblées, intégrées, décortiquées, recoupées pour pouvoir en tirer un maximum de conclusions sous formes de rapports d'analyses. Ces rapports d'analyse permettent, par exemple, de déterminer quelles sont les habitudes des clients.



Le concept de *datawarehouse* présente des similitudes avec celui de la mémoire organisationnelle. Il s'agit essentiellement de mémoriser un maximum d'informations pour améliorer l'efficacité de l'organisation. Dans le *datawarehouse*, l'utilité de l'information n'est plus spécifique à une activité, mais elle est intégrée à d'autres informations pour jouer un rôle au niveau supérieur de l'organisation.

Cependant, les SISMOs et les *datawarehouses* se différencient dans leur mode de conception. En effet, les SISMOs peuvent être considérés selon une approche *Top-Down*. L'organisation définit d'abord ses exigences en matière d'informations. Le niveau stratégique de l'organisation décide quelles sont les informations qu'il est nécessaire de mémoriser de manière à améliorer l'efficacité de l'organisation et, à partir de ces exigences, des applications sont construites. Par exemple, si une organisation décide d'effectuer de l'apprentissage sur un processus particulier, celui-ci doit être conçu d'une manière différente que s'il ne devait pas offrir d'apprentissage.

Dans le cas du *datawarehouse*, l'approche est de type *Bottom-up* car les applications sont définies et il faut reconstruire la mémoire organisationnelle à partir des données de celles-ci. Une conséquence essentielle de ceci réside dans la nature de l'information à stocker. Dans le *datawarehouse* sont reprises des informations liées essentiellement aux systèmes opérationnels. Ces éléments ne sont pas suffisants pour pouvoir gérer de la mémoire organisationnelle. Le *datawarehouse* ne mémorise pas la dynamique des processus, il n'enregistre que les conséquences de leur mise en route. Les systèmes d'information opérationnels cherchent à conserver les données de travail (mémoire sémantique) nécessaires à la réalisation d'une tâche particulière. Par exemple, pour un bon de commande, les systèmes d'information enregistrent un nom de client, un numéro de facture, ... Dans le cadre de la mémoire organisationnelle, il faut pouvoir s'intéresser au contexte de la prise de commande: comment cette commande a été passée (téléphone, papier, email, ...), pourquoi elle a été introduite (changement de fournisseur, nouveau produit, plainte, ...), quelles sont les éventuelles modifications qu'il a fallu apporter pendant la réalisation définitive du bon de commande. Toutes ces informations relatives à l'épisode peuvent permettre d'améliorer l'efficacité de l'organisation. Le *datawarehouse* enregistre des informations de gestion, un SISMO utilise ces informations de gestion et enregistre en plus les processus selon lesquels ces informations ont été générées.

Nous pouvons cependant analyser le *datawarehouse* par rapport aux exigences du modèle de Stein et Zwass (1995).

D'abord, le *datawarehouse* intègre les informations dans le temps et dans l'espace. Cependant, l'intégration dans le temps n'est pas vue comme une intégration à long terme.



Les informations sont mémorisées pour les activités présentes, ou à court terme, de l'organisation.

Le *datawarehouse*, en tant que référentiel d'informations internes et externes à l'organisation (de Linares (1996)), peut en partie répondre aux exigences du sous-système adaptatif de Stein et Zwass. En effet, en consultant les informations sur leurs clients, les organisations peuvent adapter leur comportement, car elles comprennent davantage les désirs de leurs clients. Maintenant, il est certain que la fonction adaptative ne doit pas se limiter à analyser le comportement des clients ou du marché et de réagir en conséquence. Une organisation doit aussi pouvoir analyser son propre comportement pour pouvoir le corriger et cela, le *datawarehouse* ne le permet pas vraiment.

Le *datawarehouse* permet de vérifier le degré de réalisation des buts, des objectifs de l'organisation. Cette vérification ne peut cependant pas être effectuée sur tous les buts de l'organisation car il est possible que les applications ne contiennent pas d'informations liées aux buts ou objectifs organisationnels.

Le *datawarehouse* ne répond pas, à priori, aux exigences du sous-système de Maintenance des procédures de travail de Stein et Zwass. En effet, un *datawarehouse* est un référentiel des objets directement liés à l'activité de l'organisation. Il n'enregistre pas systématiquement le comportement du personnel d'une organisation. Il est peut être possible d'inférer des caractéristiques des vendeurs à partir de leur résultats, mais ceci n'est pas le but du *datawarehouse*.

Le principal problème du *datawarehouse* est que, s'il répond aux exigences des SIS-MOs telles qu'elles ont été définies par Stein et Zwass (1995), il ne le fait pas entièrement car il n'a pas été conçu dans ce but. Un SISMO est construit de façon différente d'un système d'information traditionnel. Bien qu'il puisse travailler pour un domaine bien particulier, les informations qu'il stocke et les fonctions qui y sont associées doivent être étudiées pour améliorer le fonctionnement de toute l'organisation et pas seulement améliorer l'exécution d'une tâche particulière.

En tenant compte de ces différentes idées, nous pouvons donc dire que les systèmes d'information traditionnels supportent des éléments de mémoire organisationnelle. Malheureusement, étant donné qu'ils n'ont pas été conçus dans ce but, ils ne répondent pas à toutes les exigences que les SISMOs doivent pouvoir respecter et lorsqu'ils y répondent, c'est de façon incomplète. Dans ces conditions, nous pouvons dire qu'un SISMO n'est pas un nouveau mot pour un système d'information traditionnel mais constitue une nouvelle façon de voir l'information "utile" dans une organisation.



Nous avons présenté différents exemples de SISMOs (voir point I.9.). Cependant, il convient de remarquer que ces exemples ne satisfont pas complètement aux exigences du modèle de Stein et Zwass (1995). Ceci est évident étant donné la jeunesse de ce modèle.

### **Le sous-système Intégratif**

Les systèmes (en particulier Lotus Notes) permettent l'intégration dans l'espace et dans le temps. Cependant, cette intégration est considérée d'un point de vue essentiellement technique. Il faudrait que les systèmes mémorisent davantage d'éléments par rapport à cette intégration. Par exemple, lors de la récupération d'un processus de décision, il est intéressant d'avoir des renseignements sur l'intégration spatiale (e.g. telle personne occupe tel poste dans tel département) car ces informations sont liées au contexte de la prise de décision. Les systèmes enregistrent peu de ce genre d'information.

### **Le sous-système Adaptatif**

Les systèmes répondent en grande partie aux exigences du sous-système Adaptatif. Ils permettent de voir quels problèmes ont été rencontrés dans une organisation et comment l'organisation s'y est adaptée (gIBIS, Sibyl, CGSE, Team memory). Un moyen de s'adapter est aussi d'apprendre en posant des questions (Answer Garden) ou en consultant toutes les données, non nécessairement internes, concernant un domaine particulier (Worm Community Database).

### **Le sous-système de Réalisation de but**

Les limites importantes des systèmes commencent à apparaître à ce niveau-ci. Seuls certains systèmes (Sibyl, Team Memory) mémorisent des buts, mais ces buts ne sont pas nécessairement reliés à des objectifs organisationnels. Il est donc difficile de mettre en relation les processus qui se sont déroulés dans l'organisation avec les objectifs de celle-ci.

### **Le sous-système de Maintenance des procédures de travail**

Nous pouvons aussi remarquer des carences au niveau de ce sous-système. Les systèmes enregistrent des problèmes dans lesquels les acteurs ont été impliqués. Dans Team Memory, il est possible de récupérer des projets auxquels un acteur a participé, mais étant donné que ce système n'enregistre pas de processus dynamique, il n'est pas facile de comprendre le comportement des acteurs. Les autres systèmes (Sibyl, gIBIS, CGSE) décrivent des processus dans lesquels le personnel est impliqué, mais l'absence de requêtes

liées à la gestion des acteurs rend difficile la bonne compréhension de leurs caractéristiques.

Dès lors, nous pouvons dire que les exemples de SISMOs supportent des éléments des sous-systèmes du modèle de Stein et Zwass (1995). Malheureusement, ils ne répondent pas entièrement aux exigences de ce modèle. Cette constatation est liée à la relation entre approche *top-down* et *bottom-up*. Ces exemples sont construits de manière à satisfaire le travail de groupe, mais en ne tenant pas compte des informations qu'il serait utile de mémoriser pour augmenter l'efficacité organisationnelle.

Une critique des modèles de mémoire organisationnelle que nous avons présentés doit aussi être soulignée. En effet, ces modèles ne sont pas assez développés.

Comme nous l'avons déjà noté, le modèle de Walsh et Ungson (1991) présente essentiellement deux défauts. D'une part, l'intérêt porté aux systèmes d'information est minime et, d'autre part, ce modèle englobe virtuellement tout, ce qui rend difficile la distinction entre ce qui est mémoire organisationnelle et ce qui ne l'est pas.

Le modèle de Stein et Zwass (1995) présente les exigences des SISMOs, mais un certain nombre de points vagues subsistent. Le modèle souffre essentiellement de son manque de précision quant aux données qui doivent réellement appartenir à un SISMO, ainsi qu'à la forme des différentes fonctions affichées dans ce modèle. De la même manière, le modèle de Morrison et Weiser (1996) introduit la base mémoire d'un SISMO, mais aucun détail du contenu de cette base mémoire n'est présenté. La base mémoire peut être accédée via quatre fonctions mnémoniques (acquisition, récupération, recherche et maintenance) mais celles-ci sont peu développées. Enfin, ces modèles ne précisent pas si la situation générale d'une organisation serait plutôt un SISMO puissant gérant toute la mémoire organisationnelle, ou bien plusieurs SISMOs, chacun assigné à la mémoire d'une ou plusieurs tâches particulières.

En conclusion, les modèles actuels présentent des faiblesses quant à la description concrète du contenu (données et fonctions) des SISMOs.



## **I.11. Résumé**

---

De nos jours, les organisations doivent retenir de plus en plus d'informations. Généralement les informations retenues dans une organisation sont liées à la gestion des affaires courantes (clients, fournisseurs, commandes) et peu d'intérêt est porté aux informations liées aux processus tels que prises de décision, négociations, recherches de connaissance, etc... Pour des raisons essentiellement liées à l'efficacité, cette tendance commence à apparaître dans les entreprises. Au cours de leur histoire, les organisations se créent une mémoire : la mémoire organisationnelle. La mémoire organisationnelle constitue une source informationnelle importante dont la principale application est l'apprentissage. Il s'agit de se servir du passé pour supporter, voire améliorer, les activités du présent. La mémoire organisationnelle permet de rendre l'entreprise moins dépendante de ses membres (e.g. les rotations de personnel). S'il semble particulièrement intéressant, moyennant certaines précautions, de maintenir une mémoire organisationnelle, cette maintenance n'est pas facile. Le principal problème auquel les concepteurs de systèmes d'information supportant la mémoire organisationnelle (SISMOs) doivent faire face est la mémorisation du contexte. Lors d'une prise de décision, par exemple, il faut pouvoir mémoriser le contexte dans lequel cette décision a été prise pour pouvoir la réinterpréter plus tard. Face à ce problème, des chercheurs ont revu leurs prétentions à la baisse en ne considérant plus une mémoire pour toute une organisation, mais pour une communauté scientifique ou un groupe de travail. De cette manière, la mémorisation du contexte est plus facile car la communauté ou le groupe se partage déjà la même culture et la même base de connaissances.

Dans cette partie, nous avons voulu décrire le décor dans lequel les organisations vont être amenées à évoluer si elles désirent être compétitives. En présentant différents modèles liés à la mémoire organisationnelle, nous avons voulu relever les exigences en matière de données et de fonctions auxquelles un SISMO doit pouvoir répondre. Les quelques exemples de SISMOs que nous avons exposés permettent aussi de décrire l'allure générale de ces nouveaux systèmes.

Dans la deuxième partie de ce travail, nous proposons une extension du langage d'argumentation ARBAS. Nous transformons ARBAS de manière à ce qu'il puisse rencontrer davantage les exigences des SISMOs. Cette transformation permet de relever, de manière concrète, quelles sont les données et les fonctions qui sont supportées par un SISMO dédié aux processus de négociation.





---

## ***PARTIE II : La mémoire organisationnelle dans le cadre de la négociation***

---

### ***II.1. Introduction***

---

Dans la partie précédente, nous avons montré qu'il était intéressant de gérer une mémoire organisationnelle (I.4 et I.5), et ce, malgré les dangers que cela impliquait (I.6). Dans cette deuxième partie, nous nous concentrons sur les processus de négociation. D'abord, nous exposons différents éléments qui interviennent dans les négociations et présentons quelques systèmes d'aide à la négociation. Nous nous penchons, ensuite, sur la question de savoir en quoi il est intéressant de mémoriser des processus de négociation et d'utiliser des systèmes d'aide à la négociation.

Par la suite, nous présentons ARBAS, un outil d'aide à la négociation en soulevant quelques points critiques qui seront à l'origine d'une première version corrigée de ce système. Enfin, de manière à mieux répondre aux besoins de la mémoire organisationnelle, nous avons étendu la version corrigée de manière à donner naissance à ARBAS96, un système d'aide à la négociation supportant les aspects de mémoire organisationnelle propres à la négociation.

---

### ***II.2. Etat de l'art des systèmes d'aide à la négociation***

---

Depuis l'aube des temps, les hommes ont été amenés à négocier. Le troc constitue un exemple par lequel des personnes, généralement deux, doivent arriver à un accord pour effectuer une transaction.

<p>Kersten et al. (1991) définissent la négociation comme une forme de prise de décision impliquant activement au moins deux agents qui ne peuvent pas prendre de décisions <b>indépendamment</b> les uns des autres, et qui sont, dès lors, obligés de faire des concessions pour atteindre un compromis.</p>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Les négociations peuvent être de types différents selon qu'elles concernent des conflits entre des parties ennemies (les négociations "win-loose", "hard" ou "distributives") ou des conflits entre des parties amies (les négociations "win-win", "soft" ou "intégratives") (Nunamaker et al. (1991). Cette distinction est importante car elle représente une partie

du contexte dans lequel la négociation se déroule. La relation unissant les parties impliquées dans une négociation permet de mieux comprendre la raison de certaines décisions.

Par rapport aux conflits, Thomas (1992) présente un modèle normatif qui consiste en cinq aspects fondamentaux : la prise de conscience du conflit, les conflits normatifs, les pensées et les émotions, les intonations, le comportement et les résultats.

La **prise de conscience** d'un conflit naît généralement lorsqu'une partie prend conscience qu'une autre partie met en danger une partie ou la totalité de son champ d'action. Les conflits de buts apparaissent lorsque des parties ont des objectifs divergents sur l'issue d'un problème. Il y a aussi les conflits de jugement qui se prononcent lorsque les parties partagent le même but mais diffèrent dans la manière dont il veulent l'atteindre.

Les **conflits normatifs** sont liés au comportement attendu des différentes parties impliquées dans une négociation. Certains participants affichent un comportement illicite au cours de la négociation et peuvent, dès lors, subir des sanctions.

Les **pensées et les émotions** représentent la manière selon laquelle une partie va intervenir dans une négociation. D'une part, la pensée représente ses idées qui seront présentées selon une certaine émotion. Les situations conflictuelles sont généralement accompagnées de fortes émotions (Bui (1994)).

Les **intentions** décrivent la manière dont les négociateurs se comportent et marchandent pendant la négociation. Ils peuvent être autoritaires ou coopératifs.

Le **comportement** est la tentative de mener à bien l'une de ses intentions. Il y a toujours une différence entre les intentions et le comportement.

Le **résultat d'une négociation** est l'acceptation ou le refus d'une solution. Le résultat dépend de la justesse, de l'honnêteté de la proposition de solution. Cette justesse est, elle-même, liée à des facteurs socio-émotionnels (Bui (1994)).

Lors de ces dernières années, des systèmes supportant le travail de groupe ont été développés (e.g. GroupSystem présenté dans la première partie). Ces systèmes permettent à différentes personnes, non nécessairement situées à un même endroit, de travailler sur une tâche commune. Avec le développement des environnements de travail coopératif, sont apparus les systèmes d'aide à la négociation (*Negotiation Support System - NSS*). Ceux-ci sont souvent considérés comme faisant partie d'une classe particulière des systèmes d'aide à la décision de groupe. Les recherches en matière de négociation couvrent plu-



sieurs domaines tels que la théorie des jeux, l'intelligence artificielle, l'analyse multicritère, l'économie, les systèmes d'aide à la décision ou la psychologie cognitive.

Les systèmes qui entrent dans la catégorie des NSSs peuvent se différencier selon différents critères :

- **La structuration du système** (Chang et Han (1995)) : les NSSs peuvent être rangés selon un spectre allant des NSSs à cadres conceptuels informels, non-structurés aux NSSs à cadres conceptuels formels et structurés. Les GDSSs (*Group Decision Support System* ou systèmes d'aide à la décision de groupe) étendus pour supporter la négociation sont un exemple de NSS à cadre conceptuel informel. Au départ, ils ont été créés pour supporter les prises de décision, contrairement à d'autres systèmes qui ont, intentionnellement, été élaborés pour supporter la négociation. Le cadre conceptuel de ces autres systèmes est formalisé, car ils fournissent les moyens de capturer et de représenter explicitement les buts, les règles et les relations d'une négociation. Une manière de structurer le discours d'une négociation est d'utiliser un langage d'argumentation par lequel les participants peuvent intervenir au cours d'une négociation (Binbasioglu et al (1995)).
- **L'objectif du système** : certains NSSs s'intéressent davantage à l'évaluation des options, l'identification d'accords potentiels ou à la résolution du conflit. Ce genre de systèmes utilise des modèles mathématiques (e.g. analyse multicritère) qui délimitent l'espace des solutions acceptables par les différentes parties. Au départ, la plupart des systèmes d'aide à la négociation suivaient cette orientation. Dernièrement, d'autres NSSs se sont plutôt concentrés à offrir des moyens pour augmenter la qualité des propositions de solutions. Chang et Han (1995) voient cette distinction sous l'angle quantitatif/qualitatif. Les éléments tels que les buts, les alternatives, les conséquences sont considérés comme des aspects quantitatifs d'une négociation. Pour ce qui est des éléments qualitatifs, Chang et Han (1995) relèvent, entre autres, l'argumentation et les interactions.
- **Le domaine d'utilisation du système** : certains NSSs se limitent aux négociations d'un certain type (e.g. négociation entre seulement deux parties ou négociation salariale), tandis que d'autres permettent de négocier quelles que soient les caractéristiques de la négociation.

- **Le degré d'intelligence du système** : au cours d'une négociation, les NSSs peuvent afficher un comportement actif (intelligent) ou un comportement passif. Lorsque le NSS est passif, son support se limite à enregistrer et récupérer les informations d'une négociation suivant la demande d'un utilisateur du système. Les systèmes actifs, quant à eux, prennent des initiatives pour intervenir dans le cours de la négociation. Ils peuvent générer des propositions ou des arguments ainsi que prévenir les participants à la négociation des dangers auxquels ils risquent d'être confrontés (Sycara (1993)).
- **L'aspect évolutif du système** : au cours d'un problème de négociation, la perception des acteurs évolue, ce qui entraîne une modification de la structure de ce problème de négociation. Ces différentes modifications de structure sont à l'origine du concept de représentation restructurable (Kersten et al. (1991), Shakun (1991) et Sycara (1991)). Sycara (1991) définit la restructuration comme étant un processus changeant dynamiquement la structure du problème de négociation jusqu'au moment où un accord survient. Certains systèmes supportent la restructuration des problèmes tandis que d'autres ne la supportent pas.

Nous présentons maintenant des systèmes d'aide à la négociation existants. Ces systèmes se différencient selon les caractéristiques que nous venons d'énoncer (voir Tableau II.1).

	Structuration du système	Objectif du système	Domaine d'utilisation	Degré d'intelligence	Restructuration
Co-op	Faible	prise de décision	Quelconque	moyen (analyse multi-critère)	faible
Negotiator	Forte	négociation	Quelconque	moyen (analyse multi-critère)	moyenne
Negoplan	Forte	négociation	Quelconque	élevé	Forte
Persuader	Forte	négociation	Négociation syndicat/direction	Très élevé	Forte
Ibis et Sibyl	Forte et Souple	argumentation	Quelconque	moyen à élevé	Forte

Tableau II.1. : Etat de l'art des systèmes d'aide à la négociation



## Co-op

Co-op (Bui et Jarke (1986)) est un GDSS permettant la prise de décision de groupe multi-critères et coopérative. Co-op obéit à six étapes dans la résolution d'un problème de groupe multi-critères :

- la définition du problème, portée par le groupe collectivement.
- la définition de la norme du groupe. Cette norme identifie les membres du groupe et leur assigne des mots de passe individuels.
- la détermination des priorités des critères d'évaluation. Cette détermination des priorités peut se faire selon trois modes : *pooled* (les membres assignent collectivement un vecteur commun de priorités), *séquentiel* (les membres, en fonction de leur expertise, assigne des priorités à un sous-ensemble de critères, *agrégé* (chaque membre assigne des poids individuels qui sont ensuite agrégés).
- la sélection des alternatives individuellement. Co-op permet à chaque preneur de décision d'étudier les alternatives individuellement à l'aide de son ou de ses modèles de décision multi-critères.
- la sélection des alternatives en groupe. Traitement informatique des résultats du groupe déduits, par des techniques d'agrégation, à partir des résultats individuels.
- la recherche de consensus et la négociation. Si l'unanimité n'est pas atteinte, des algorithmes de recherche de consensus peuvent intervenir dans cette dernière étape. Si aucun accord ne survient, les preneurs de décision peuvent réviser la définition du problème. Co-op supporte plusieurs méthodes de recherche de consensus et de propositions de concessions.

La dernière étape nous intéresse principalement car elle concerne la négociation. Co-op offre des moyens de gérer la négociation mais ces moyens sont établis dans le cadre d'un GDSS. Co-op peut donc se classer parmi les NSSs informels. L'objectif du système est d'arriver à une solution et le moyen utilisé pour y arriver est l'application de modèles mathématiques. En ce qui concerne l'intelligence du système, le rôle de Co-op se limite à la recherche de consensus par l'application des modèles d'analyse multi-critères. Co-op ne propose pas, par exemple, une solution au conflit sur base de recherche de problèmes passés. Le contrôle de l'aspect évolutif est en partie assuré dans Co-op puisque lorsque

les preneurs de décision n'arrivent pas à une solution, ils peuvent redéfinir le problème. Il nous faut remarquer tout de même que Co-op ne garde pas de traces des enchaînements des différentes redéfinitions du problème.

Nos critiques à l'égard de Co-op peuvent sembler sévères, mais ces jugements sont surtout influencés par le fait que Co-op a d'abord été élaboré pour supporter la prise de décision de groupe.

### **Negotiator**

Negotiator (Bui et Shakun (1995)) est un NSS basé sur un cadre conceptuel d'élaboration de systèmes évolutifs (Shakun (1991)). Dans ce cadre conceptuel, un problème de négociation implique les ensembles d'éléments suivants : des valeurs, des buts opérationnels, des décisions ou des actions prises pour atteindre ces buts, des critères d'évaluation de ces décisions, des préférences individuelles et des préférences de groupe basées sur les préférences individuelles. Un problème est représenté par des relations évolutives entre ces 6 ensembles d'éléments. Shakun (1991) présente deux hiérarchies de relations entre ces éléments. La première représente un cadre conceptuel permettant de rechercher dynamiquement la définition du problème général à traiter : les relations buts/valeurs et contrôles/buts. La seconde hiérarchie, liée au processus de recherche de solution, implique 5 espaces de solutions évolutifs (décisions, buts, critères, préférences individuelles et préférences de groupe).

Negotiator permet aux différentes parties impliquées dans la négociation de voyager dynamiquement à travers ces hiérarchies et d'essayer de diminuer leurs différences de vues.

Negotiator utilise un modèle mathématique pour la recherche d'une solution optimale par rapport aux attentes des parties impliquées dans la négociation.

Par rapport aux critères que nous avons relevés plus haut, Negotiator se situe en bonne position. Il a été créé pour la négociation et sa structure y est donc liée. Ensuite, Negotiator supporte la restructuration des problèmes grâce à son cadre conceptuel évolutif. Negotiator utilise un modèle de décision multi-critères pour rechercher un accord. Negotiator ne s'occupe pas d'améliorer la phase de génération des décisions sur base d'anciennes décisions, mais il se limite à les traiter.

Negotiator peut traiter différents types de négociations. Enfin, même si Negotiator tient compte de certains aspects qualitatifs (caractère évolutif de la négociation) ce sont



surtout les aspects quantitatifs qui y sont supportés. Il est donc difficile de l'utiliser pour des négociations dont les caractéristiques ne sont pas mesurables.

### Negoplan

Negoplan (Kersten et al. (1991) et Kersten (1995)) est un système d'aide à la négociation qui supporte la modélisation de problèmes de décision complexes, dynamiques et séquentiels.

Negoplan est un système expert implémenté en Prolog. Il supporte aussi bien les aspects quantitatifs que qualitatifs de la négociation. Différentes constructions clés sont utilisées dans Negoplan pour représenter, simuler et supporter les processus de négociation (Kersten (1995)) : des règles de la forme *goal*  $\leftarrow$  *subgoals & issues* (e.g. partir en vacances  $\leftarrow$  aller à la mer & faire du camping), des meta-faits qui décrivent les points de discussion, les buts et les informations à propos du monde, et les attributs liés au comportement et les résultats des actions, des relations entre ces meta-faits et des meta-règles qui assurent la restructuration du problème. Negoplan a été structuré de manière à répondre aux exigences des négociations, de plus, en tant que système expert, Negoplan possède une certaine dose d'intelligence. Il supporte aussi la restructuration des problèmes (Kersten et al. (1991)). Negoplan peut, via son système de règles, supporter n'importe quel genre de négociation. Negoplan est certainement un des systèmes d'aide à la négociation les plus solides actuellement.

### Persuader

Persuader (Sycara (1991), Sycara (1993)) est un système d'aide à la négociation qui agit par les mécanismes de case-based reasoning. Le case-based reasoning utilise la connaissance de prises de décisions antérieures traitant du même sujet (cas) et propose des solutions à un nouveau cas (la prise de décision courante) (Sycara (1993)). Persuader utilise des modèles mathématiques pour la recherche de compromis pouvant satisfaire les parties impliquées.

Le modèle de Persuader est capable :

- de générer des propositions initiales,
- de générer des contre-propositions qui rapprochent les différences entre les participants,

- de prendre en considération les changements des circonstances, la raison des convictions des participants
- de générer des justifications et des arguments pour supporter ou contrer des décisions.

La force de Persuader est de pouvoir apprendre par expérience. Non seulement Persuader peut apprendre à partir des cas à succès, mais il peut aussi apprendre à partir des échecs. Du point de vue des échecs, il intervient de deux manières. D'une part, Persuader contrôle que les erreurs qui se sont produites dans le passé ne se reproduisent pas dans le futur et, d'autre part, lorsqu'un échec survient, Persuader propose une solution (éventuellement modifiée) qui a été appliquée dans le passé.

Par rapport aux critères que nous avons proposés plus haut, Persuader se situe en très bonne position. D'abord, Persuader permet de restructurer les problèmes (Sycara (1991)). Ensuite, ce système a été conçu dans le but de répondre aux besoins des négociations. Pour l'instant, Persuader se limite aux conflits direction/syndicats, mais il est possible de l'étendre à d'autres domaines. Le problème est qu'il doit tout de même être lié à un domaine particulier car la plupart des concepts de ce domaine doivent être mémorisés. Cette lacune rend le système relativement rigide.

Une autre catégorie de systèmes d'aide à la négociation cherche à structurer, à l'aide d'un langage d'argumentation, les discussions qui s'opèrent pendant une négociation. Cette vision des choses est basée sur la théorie de Toulmin (1958) qui suggère d'utiliser un **langage d'argumentation** afin de structurer les arguments, les raisons, les preuves et les hypothèses. Parmi les systèmes supportant un langage d'argumentation, nous citerons (g)Ibis et Sibyl, tout deux présentés précédemment (voir point I.92.). Chang et Han (1995) ont étendu IBIS de manière à augmenter ses capacités pour, d'une part, exprimer l'aspect processus du discours, et d'autre part, pour supporter l'évaluation dynamique des arguments. Cette extension (*Dialectic Extension to IBIS*) permet de représenter toute négociation. ARBAS est un autre système qui est basé sur un langage d'argumentation. Ce système fait l'objet de notre travail et sera détaillé plus loin. L'avantage d'un langage d'argumentation est qu'il est souple dans son utilisation. Il améliore la communication qui est reconnue comme un problème clé du processus de négociation (Sycara (1993), Bui (1994), Bui et Shakun (1995)).

La plupart des systèmes actuels présentent des lacunes en matière de protocole de négociation (Chang et Woo (1994)). Le protocole de négociation est important car il fournit une structure au processus de négociation et permet, dès lors, d'identifier les par-



ties automatisables et non-automatisables. Le principal objectif d'un protocole de négociation n'est pas seulement de guider le comportement des agents humains, mais aussi d'améliorer la capacité d'intervention d'un ordinateur dans une négociation (Chang et Woo (1994)). Nous notons tout de même que les protocoles actuels sont limités à des négociations impliquant seulement deux parties.

Une classe évoluée des NSSs sont les systèmes de support cognitif (*cognitive support systems*). Ils doivent être capables, selon Kersten et Cray (1994), de :

- fournir les moyens de structurer les problèmes.
- de supporter le problème de négociation.
- d'intégrer les représentations quantitatives et qualitatives.
- de supporter les deux approches de la prise de décision, à savoir : la généralisation et l'analogie.
- de fournir des descriptions des alternatives.

Par rapport aux systèmes existants, Kersten (1995) conclut qu'aucun d'entre-eux ne répond actuellement à ces exigences.

En conclusion, les systèmes d'aide à la négociation se sont développés au cours de ces dernières années. Des progrès ont été constatés dans la manière de structurer les problèmes de négociation et dans la façon dont les systèmes interviennent en cours de négociation. L'introduction de langages d'argumentation a rendu plus souple la communication dans les négociations ainsi que la manipulation de leurs informations. Par rapport à la mémoire organisationnelle, il est certain que ces systèmes en supportent des éléments. Seulement, étant donné qu'ils n'ont pas été élaborés de manière à répondre aux exigences auxquelles les SISMOs doivent pouvoir répondre, ils se trouvent quelque peu limités par rapport à ce concept.

### ***II.3. Pourquoi mémoriser les processus de négociation ?***

---

Dans cette section, nous montrons, dans quelle mesure, la mémorisation des processus de négociation est bénéfique pour une organisation. Indirectement, nous justifions aussi l'utilité des systèmes d'aide à la négociation. Notre démarche consiste à reprendre des idées liées à la mémoire organisationnelle et à les fixer dans le cadre restreint aux proces-

sus de négociation. Un processus de négociation peut être décomposé en quatre phases : la pré-négociation, la négociation proprement dite, la clôture de la négociation et la post-négociation. Nous exposons de quelle manière la mémorisation des processus de négociation permet d'améliorer chacune de ces phases.

### **La pré-négociation**

La phase de pré-négociation a pour objectif d'établir une définition complète du problème de négociation à résoudre. D'une manière générale, une définition de problème comprend l'événement déclencheur du problème, les acteurs impliqués dans ce problème ainsi que les buts qu'ils espèrent voir se réaliser. La définition d'un problème peut aussi comprendre le protocole de négociation qui régit les interventions des participants au cours de la négociation. **La mémorisation des processus de négociation permet d'améliorer la phase de pré-négociation d'un processus.**

Lorsque des résolutions de problèmes de négociation appartiennent déjà à la mémoire organisationnelle, il est, par exemple, possible de les consulter pour voir quels sont les terrains de prédilection des acteurs. Certains acteurs peuvent être habitués aux négociations salariales tandis que d'autres peuvent avoir été impliqués dans des négociations avec les clients. Ces informations sont utiles lorsqu'il s'agit de choisir l'acteur qui devra défendre les intérêts de l'organisation ou d'un département de celle-ci. La sélection d'un acteur peut encore être influencée par l'évaluation de son comportement au cours de ses précédentes négociations.

Généralement, les acteurs doivent émettre des buts lors de la définition d'un problème. La mémoire organisationnelle peut aider chaque acteur dans cette tâche car elle offre le moyen de consulter tous les buts qui ont déjà été émis lors de précédentes négociations. Ainsi, les acteurs (ou le système) peuvent analyser si un but risque d'apporter ou d'empêcher une solution. Dans la même optique, il est possible de percevoir quels sont les protocoles de négociation les plus aptes à apporter la résolution d'un conflit.

Au cours de cette phase de pré-négociation, la consultation des processus de négociation mémorisés peut aussi permettre l'émission de la proposition de solution initiale. Il s'agit de se servir du passé pour offrir un premier point de discussion aux acteurs.

### **La négociation proprement dite**

Dès qu'une définition du problème de négociation a été énoncée, les acteurs vont pouvoir s'exprimer pour défendre leurs intérêts. Au cours de cette étape, la mémoire organisationnelle assume deux rôles principaux :



**1. Fournir à chacune des parties impliquées dans un processus de négociation, un meilleur support informationnel par rapport à ce processus.**

- Meilleure compréhension des acteurs par rapport à l'évolution de leur propre comportement.

Au cours d'une négociation, les acteurs émettent des propositions d'actions pour résoudre le problème auquel ils sont confrontés. Au départ de la négociation, les buts et les critères des acteurs sont vagues. Les acteurs veulent aboutir à un résultat mais ne savent pas encore exactement comment ils doivent s'y prendre. De même leurs arguments développés en début de négociation ne sont pas encore persuasifs. La mémorisation du processus de négociation offre aux acteurs un moyen de mieux comprendre la manière dont ils ont agi au cours de la négociation.

- Meilleur contrôle de la logique des attitudes des acteurs.

Les acteurs, pouvant analyser leur propre comportement au cours de la négociation, ont aussi la possibilité d'étudier la logique de leurs attitudes. Ils peuvent saisir si leurs actions suivent un fil conducteur ou si elles sont émises de manière marginale, illogique. En effet, pour bien faire, les acteurs doivent s'assurer que leur proposition de solution fasse progresser la négociation.

- Meilleure compréhension des acteurs par rapport aux comportements des parties adverses.

Une meilleure compréhension des comportements des parties adverses permet à un acteur d'analyser le chemin sur lequel ses adversaires tentent de le conduire. Si, ce chemin le séduit, c'est-à-dire va dans la direction qu'il espère, il émettra des propositions de solutions s'accordant avec les attitudes des parties adverses. Par contre, si l'acteur sent que les attitudes adverses vont à l'encontre de ses intérêts, il réagira de manière à se défendre, au risque de bloquer la négociation.

- Meilleure intégration des nouveaux acteurs d'une négociation.

Au cours d'une négociation, il arrive que des acteurs entrent en jeu soit pour remplacer d'autres acteurs, soit en tant que nouvel intervenant. La mémorisation du processus de négociation permettra au nouvel arrivant de comprendre plus rapidement le contexte dans lequel il a atterri. Il pourra analyser les comportements de son éventuel prédécesseur ainsi que ceux des parties adverses.

- Meilleur support pour rechercher des arguments plus persuasifs.

En scrutant les négociations passées de l'organisation, il est possible de retrouver les arguments qui ont été particulièrement convaincants et d'en tirer profit pour la négociation courante. La qualité des propositions émises par les acteurs est augmentée grâce à l'ensemble des informations disponibles.

- Moyen d'éviter les erreurs passées.

La mémorisation des processus de négociation permet, lorsqu'un acteur émet une proposition de solution, de contrôler que cette proposition n'est pas une erreur. Ce contrôle peut se faire de manière automatique ou semi-automatique en analysant la mémoire organisationnelle.

- Moyen offert aux acteurs pour convaincre les parties adverses de leur honnêteté.

Le principe d'un système de négociation consiste à diffuser les informations entre les acteurs impliqués dans un processus de résolution de conflit. De ce fait, chaque acteur joue cartes sur table. Les acteurs présentent des propositions de solutions et celles-ci doivent être supportées par des arguments sous peine de ne pas être prises en considération par les autres acteurs. Les systèmes d'aide à la négociation réclament une justification des actions des participants.

## **2. Fournir un meilleur support au secrétaire.**

La mémorisation d'un processus de négociation offre la possibilité à tous les acteurs, et en particulier au secrétaire, d'avoir une vue d'ensemble sur la tentative de résolution du conflit. Ce support est particulièrement utile pour le secrétaire car il lui donne un moyen de mieux comprendre le comportement de chacun et ainsi de repérer les idées convergentes, sources de solutions. Dans le cas d'une situation de blocage, le secrétaire



pourrait proposer un tour de table parmi les participants pour que chacun revoie ses prétentions à la baisse, par exemple.

### **La clôture de la négociation**

La clôture de la négociation s'occupe de la terminaison de la négociation. Les conditions de terminaison de la négociation peuvent être multiples : résolution du conflit (proposition acceptée par tous les acteurs), temps imparti écoulé (abandon de la négociation), nombre maximum de propositions émises dépassé (abandon de la négociation), majorité qui supporte une proposition (solution imposée), etc. ...

Dans cette phase, la mémoire organisationnelle liée à la négociation peut **fournir un meilleur support au secrétaire et aux acteurs pour la production des rapports et pour la ratification de l'accord.**

Toutes les informations relatives à la négociation sont immédiatement centralisées dans le système informatique. De cette manière, l'édition du rapport final peut être automatisée. Ce rapport présentera les différents acteurs de la négociation, ainsi que leurs diverses interventions. La ratification de l'accord est composée de deux phases : la rédaction et le dépôt des signatures. La rédaction de l'accord, lorsque celui-ci est lié à la mise en oeuvre d'une proposition, peut être automatisée. Pour une raison essentiellement légale, le dépôt des signatures sur document "papier" ou électronique ne peut, quant à lui, se faire que d'une manière manuelle.

### **La post-négociation**

Cette phase ne fait pas, à proprement parlé, partie du processus de négociation. Il s'agit d'une phase qui n'est déclenchée que lorsque la négociation est terminée. Au cours de la post-négociation, la mémoire organisationnelle **fournit un moyen d'évaluation de l'efficacité de la négociation.** Le déroulement de la négociation a été mémorisé par l'organisation, et il est ainsi possible d'étudier l'attitude de chacun des acteurs, de découvrir les succès encourus, de repérer les erreurs commises, etc. ....

Nous avons présenté différentes raisons pour lesquelles la mémorisation des processus de négociation est utile, voir nécessaire. Tableau II.2. résume les arguments supportant la mémorisation des processus de négociation.

<b><u>Phases</u></b>	<b><u>Support des Systèmes d'aide à la négociation</u></b>
<b>Pré-négociation</b>	<ul style="list-style-type: none"> <li>• Amélioration de la définition du problème</li> </ul>
<b>Négociation</b>	<ul style="list-style-type: none"> <li>• Emission d'une première proposition de solution</li> <li>• Meilleure compréhension des acteurs par rapport à l'évolution de leur propre comportement.</li> <li>• Meilleur contrôle de la logique des attitudes des acteurs.</li> <li>• Meilleure compréhension des acteurs par rapport aux comportements des parties adverses.</li> <li>• Meilleure intégration des nouveaux acteurs d'une négociation.</li> <li>• Meilleur support pour rechercher des arguments plus persuasifs.</li> <li>• Moyen offert aux acteurs pour convaincre les parties adverses de leur honnêteté.</li> <li>• Meilleur support au secrétaire</li> </ul>
<b>Clôture de la négociation</b>	<ul style="list-style-type: none"> <li>• Meilleur support au secrétaire et aux acteurs pour la production des rapports et pour la ratification de l'accord.</li> </ul>
<b>Post-négociation</b>	<ul style="list-style-type: none"> <li>• Moyen d'évaluation de l'efficacité de la négociation.</li> </ul>

Tableau II.2. : Pourquoi mémoriser les processus de négociation ?



Nous avons montré l'intérêt de la mémorisation des processus de négociation. Cette mémorisation n'implique pas nécessairement l'utilisation d'un système d'aide à la négociation. Nous nous occupons de présenter, dès à présent, l'intérêt des systèmes d'aide à la négociation (NSSs).

Les NSSs permettent d'éliminer les émotions (Bui (1994)). Les parties ne sont plus directement confrontées les unes aux autres, mais passent par l'intermédiaire du système informatique. Avec l'intervention d'un NSS, les fortes personnalités disposent de moins d'atouts pour pouvoir s'exprimer avec conviction (e.g. élévation de la voix pour impressionner les parties adverses).

Selon une perspective normative et socio-émotionnelle, l'utilisation d'un NSS influence la dynamique de négociation en créant une atmosphère qui règle les échanges entre les différentes parties (Bui (1994)). Bui (1994) propose 5 hypothèses de développement de NSSs :

- La consommation de ressources (e.g. temps, effort) sera moindre pour des parties utilisant un NSS que pour celles qui n'en utilisent pas. Cette épargne de ressources est perçue à long terme car, au départ, les parties non habituées au système risquent peut-être de consommer davantage de ressources.
- La qualité de la décision sera plus grande pour des parties utilisant un NSS que pour celles qui ne le font pas. L'utilisation de systèmes de classement des points de discussion permet de mettre en évidence les facteurs clés.
- La perception de l'honnêteté est améliorée tant au niveau du processus de négociation qu'au niveau de son résultat. Les parties jouent cartes sur table.
- La satisfaction du résultat sera plus grande pour des parties utilisant un NSS que pour celles qui ne le font pas. L'utilisation de modèles d'optimisation est la principale raison de ce plus grand degré de satisfaction.
- Les relations de travail seront meilleures pour des parties qui utilisent un NSS que pour celles qui ne le font pas. La négociation précède généralement un travail collectif pour la mise en place de l'accord. La négociation se déroulant sous de meilleures conditions avec un NSS, détériore moins les relations de travail qui unissent les différentes parties.

## II.3. Présentation générale d'ARBAS

ARBAS (*Action-Resource Based Argumentation Support*) est un langage d'argumentation qui a été créé initialement pour servir d'aide à la négociation (Binbasioglu et al. (1995), Bui et al. (1995)). Un problème de décision est considéré comme une situation dans laquelle le preneur de décision doit identifier quelles sont les actions à entreprendre et quelles sont les ressources affectées par ces actions. Dans un contexte de négociation, ARBAS permet à des membres d'un groupe d'échanger des informations, des opinions et des propositions dans le but de résoudre le ou les problèmes auxquels ils sont confrontés.

### II.3.1. Cadre d'utilisation d'ARBAS

ARBAS est un outil qui peut s'intégrer dans des contextes intra ou inter-organisationnels. Une négociation intra-organisationnelle est une tentative de résolution de conflit dans laquelle, par exemple, différents membres d'une organisation sont impliqués. La négociation inter-organisationnelle représente une situation délicate à gérer entre plusieurs organisations. Dès lors, les termes *acteurs* et *parties* que nous utiliserons dans la suite du texte doivent être compris comme étant des participants à la négociation qu'il s'agisse de personnes, de départements ou d'organisations.

Bui et al. (1995) propose de représenter le cadre conceptuel d'ARBAS par une pyramide à trois niveaux (voir Figure II.1.)



Figure II.1. : Pyramide d'ARBAS

**Le premier niveau ou niveau inférieur :**



Le niveau le plus bas de la pyramide est constitué d'un ensemble d'outils qui servent de support aux prises de décisions de l'individu. Cette partie de la pyramide est privée dans la mesure où chaque acteur prend ses décisions selon la manière qui lui semble être la plus adéquate.

#### **Le deuxième niveau ou niveau moyen :**

Ce niveau comprend la planification (élaboration de scénario) et la prise de décision de l'organisation. Ce niveau s'appuie sur le niveau inférieur en utilisant les outils qui permettront d'augmenter la qualité des prises de décision. De part la nature de son contenu, le deuxième niveau est un niveau privé. Les informations qu'il contient ne concernent que la partie qui les a générées. Les idées construites individuellement au niveau inférieur sont exportées dans un contexte collectif.

#### **Le troisième niveau ou niveau supérieur :**

Ce niveau contient le langage d'argumentation ARBAS.

Dans le cadre d'une négociation, dès qu'une partie a décidé une proposition de solution au conflit (niveau deux de la pyramide), elle peut la rendre publique à l'aide d'ARBAS (pour autant qu'elle en ait l'autorisation). Dans le cadre plus large de la mémoire organisationnelle, ARBAS va récupérer des informations qui peuvent être utiles à la planification et aux prises de décision. Le niveau supérieur est un niveau public de communication.

La pyramide représente le cadre d'utilisation théorique d'ARBAS. Il nous semble important de le présenter même si nous ne l'utilisons plus par la suite.

### ***II.3.2. Les règles d'accès d'ARBAS***

ARBAS offre la possibilité de créer des informations (en émettant une nouvelle proposition, par exemple) et la possibilité de consulter les informations. Un contrôle des accès à ces informations doit être établi afin d'en maintenir la sécurité (voir les attributs d'un SISMO).

ARBAS propose trois niveaux d'accès :

- **L'accès Public :** Tout le monde a le droit d'accéder (via des requêtes) aux informations mémorisées par ARBAS.

- **L'accès Autorisé** : Seuls les acteurs autorisés peuvent émettre des propositions lors de la résolution d'un conflit.
- **L'accès Privé** : Seul l'acteur à l'origine d'un élément d'une négociation (problème, proposition, argument, contre-argument, critique) peut le modifier ou le supprimer.

### II.3.3. Architecture générique d'ARBAS

ARBAS est un système qui permet à différentes parties d'échanger des informations. Dans une organisation, différents types d'utilisation du système ARBAS peuvent être envisagés :

- Utilisation individuelle : chaque membre de l'organisation dispose d'un accès au système ARBAS à partir de sa propre machine.
- Utilisation partagée : les membres de l'organisation se partagent la machine munie du système ARBAS. Tour à tour ils peuvent disposer du système.
- Utilisation groupée : le système sert de support à un groupe de travail sans désignation particulière de son utilisateur.

Chaque organisation disposant d'un système ARBAS peut communiquer avec toute autre organisation disposant du même système. Les liens entre des organisations peuvent être des liens permanents ou *ad-hoc*. Les liens *ad-hoc* représentent des liens qui sont établis dans le cadre d'une seule négociation par exemple. Leur disparition est liée à la terminaison de la négociation.

La matrice (espace, temps) nous offre un moyen de présenter les contextes qui peuvent être supportés par ARBAS (Figure II.2.).

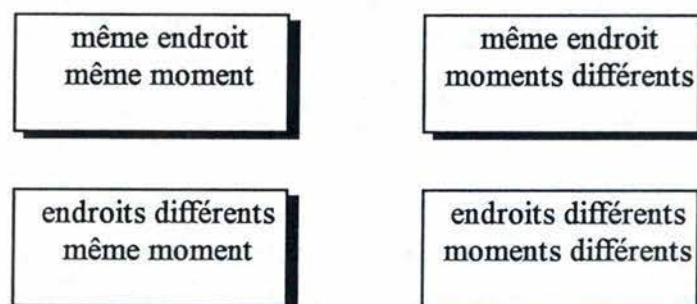


Figure II.2. : Matrice (espace, temps)



ARBAS peut supporter chaque élément de cette matrice.

En effet, ARBAS mémorise les processus de négociation de telle manière qu'un acteur de la négociation peut intervenir quand bon lui semble. Il est parfois obligé d'émettre une proposition en respectant une échéance (dans le cas d'un tour de table), mais sinon il peut faire connaître son opinion dans une négociation quand il le désire pour autant qu'il y soit autorisé. Nous reviendrons en détail plus tard sur cette notion de tour de table. A propos de la distribution géographique des acteurs, nous rappellerons que, dans ARBAS, les participants ne doivent pas nécessairement être situés au même endroit.

## ***II.4. ARBAS - outil d'aide à la négociation***

---

ARBAS est un langage d'argumentation et, comme tout langage, il possède une lexicographie, une syntaxe et une sémantique que nous présentons dans cette section. Nous exposons aussi un schéma Entité-Association d'ARBAS qui synthétise les différents éléments de ce système.

De manière à offrir une explication claire d'ARBAS, nous proposons ensuite, un exemple d'application de ce système dans le monde du développement de logiciels (Bui et al. (1995)).

ARBAS connaît certaines limites que nous relevons aussi dans cette section. Face à ces limites, nous suggérons des solutions qui constituent la première étape d'extension d'ARBAS représentée à l'aide d'un nouveau schéma Entité-Association.

### ***II.4.1. La lexicographie d'ARBAS***

La lexicographie définit le vocabulaire de l'argumentation. Un argument est construit exclusivement à partir de mots nécessaires pour un discours basé sur le modèle action-ressource. ARBAS adopte le vocabulaire suivant (Bui et al. (1995)):

Etant donnée une action qui doit être accomplie en commun (*Action\_Name*) et les ressources nécessaires pour l'accomplir (*Resource\_Name*), chacun des membres de la négociation (*View\_Owner*) exprime sa position (*View\_Position*) avec une intonation personnelle (*View\_Intonation*). La position est justifiée par un discours d'arguments

(*View\_Justification*) liés aux impacts possibles de l'action critiquée sur les ressources (*Resources\_Name*; *Resources\_Type*). Les ressources sont aussi caractérisées par leur quantité nécessaire et disponible (*Resource\_Quantity*; *Resource\_limit*).

La notation du lexique est la suivante :

View_owner:	$Z = \{ z \mid z \text{ est le nom d'un membre de la négociation } \}$
Activity_Triplet:	$A = \{ RN, RT \mid X \mid RN', RT' \}$
Action_Object:	$O = \{ o \mid o \text{ est un objet utilisé pour décrire l'action } \}$
Action_Name:	$X = \{ x(o) \mid x \text{ est le nom de l'action } \}$
Resource_Name:	$RN = \{ rn \mid rn \text{ est le nom d'une ressource } \}$
Resource_Type:	$RT = \{ Input, Output \}$
Resource_Limit:	$RL = \{ r, l \mid r \geq 0; l \text{ est la limite supérieure et inférieure de la ressource} \}$
View_Intonation:	$I = \{ Sentiment, Opinion, Belief, Conviction, Persuasion \}$
View_Position:	$P = \{ Support, Oppose, Don't Care \}$
View_Justification:	$J = \{ n \mid n \text{ est la justification de la position dans un texte libre } \}$
Proposed_Move:	$M = \{ Drop, Implement, Modify, Other \}$
Time_Stamp:	$T = \{ t \mid t \text{ est une estampille appliquée sur chaque activité entre le début et la fin de la négociation} \}$

## II.4.2. La syntaxe d'ARBAS

Une fois le lexique défini, nous présentons les règles ou la syntaxe que doit respecter toute combinaison de mots dans ARBAS (Bui et al. (1995)).

- Quand "un acteur exprime sa vue sur les ressources qui le concernent", une "*View\_On\_Resource*" peut être définie comme ceci :

$$V_{On-Resource}(Z, RN, RT, RL)$$



- Une "View\_On\_Action" peut être construite pour décrire la situation dans laquelle "un acteur, avec sa ressource, propose une action à un moment t":

$$V_{\text{On-Action}}(V_{\text{On-Resource}}, A, T)$$

- La position sur la vue et son intonation peuvent être construites comme ceci:

$$V_{\text{Position}}(V_{\text{On-Action}}, I, P)$$

- Une action recommandée peut être spécifiée en utilisant M et J qui suggère une activité appropriée :

$$V_{\text{What-move}}(V_{\text{Position}}, M, J)$$

### II.4.3. La sémantique d'ARBAS

Même si des phrases définies dans ARBAS sont correctes du point de vue de leur syntaxe, cela ne signifie pas qu'elles le sont sémantiquement. Nous relevons, ci-après, cinq contraintes qui assurent une correction minimale de la sémantique des phrases (Bui et al. (1995)).

- *Argumentation Logic*: Une position, P, devrait être suivie par un changement logique, M:

$$S = \{ (p,m) \mid (\text{Support, Implement}), (\text{Support, Modify}), (\text{Oppose, Drop}), (\text{Oppose, Modify}), (\text{Oppose, Other}), (\text{Don't Care, } m^*) \}$$

- *Operative Constraints*: Une règle de transition est définie pour assurer la continuité de l'argumentation. Une nouvelle action,  $x_t$ , au temps  $t+1$  sera proposée quand, au temps  $t$ ,  $s_t \in S'$ :

$$\exists x_{t+1}, \in X, s_t \in S' = \{ (p,m) \mid (\text{Support, Modify}), (\text{Oppose, Modify}), (\text{Oppose, Other}) \}$$

- L'argumentation et la génération d'une nouvelle action  $x$  déclenchée par  $m$  ne doit pas, en théorie, nécessairement mener à une terminaison du problème. Une contrainte imposant une limite de temps pour terminer l'argumentation peut être posée :

$$\exists V(t), t \leq t' \quad \text{ou } t' \in T, t' \text{ est un temps limite.}$$

Une autre terminaison peut être basée sur la génération d'un nombre maximum d'actions:

$$\exists X, \delta(X) \leq c' \quad \text{ou } \delta \text{ est le "numéro" de l'action } X \text{ et } c' \text{ un entier positif.}$$

Le troisième type de terminaison est défini par une limite de consensus. La limite  $s'$  est une valeur à laquelle les acteurs se sentent satisfaits :

$$\exists x' \in X, \forall (z, x', p, i) \quad \text{alors } \Sigma(p, i)_z \geq s' \quad \text{est rencontré pour tout } z \in Z.$$

#### II.4.4. Le schéma graphique d'ARBAS

Nous présentons un schéma graphique (Figure II.3.) afin de synthétiser les différents concepts présents dans ARBAS. Les éléments présents dans ce schéma ont été traduits en français de manière à faciliter sa lecture.

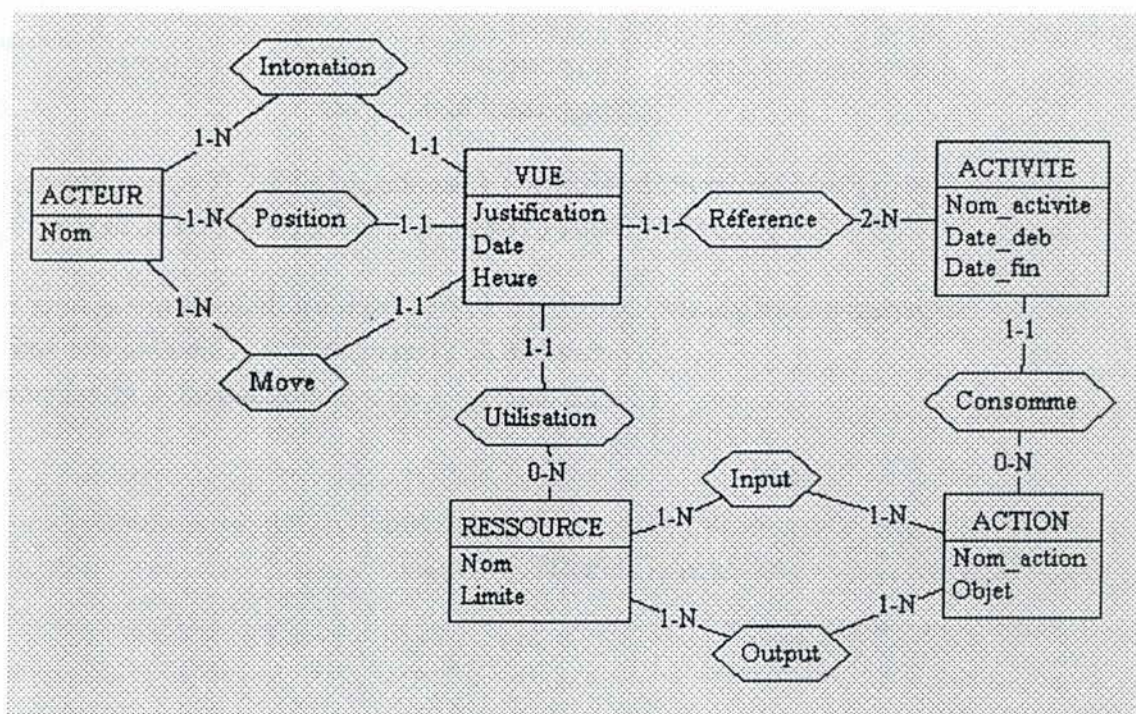




Figure II.3. : Le schéma graphique d'ARBAS

#### II.4.5. Exemple

De manière à éclaircir les esprits encore embrumés, nous présentons un exemple de négociation supportée par ARBAS. Cette présentation est rendue possible grâce à une version d'ARBAS implémentée dans l'environnement MS-Windows. L'exemple que nous présentons a été développé dans la littérature (Bui et al. (1995)).

Le problème qui nous occupe est une négociation qui implique trois parties (un département des ventes, un département des finances et une équipe de développeurs de logiciels) et qui concerne la mise à jour d'un système de gestion de commande de produits.

Le problème qui unit les différentes parties provient principalement du fait que le département des ventes et celui des finances ont des vues différentes quant aux mises à jour à effectuer sur le système. De plus, l'équipe des développeurs n'a ni le temps, ni les ressources nécessaires pour satisfaire aux exigences des deux départements.

Le département des finances, désirant disposer de plus de contrôle sur les coûts liés aux inventaires, réclame l'ajout d'un champ dans le système de gestion de commande. La Figure II.4. montre le triplet d'activité (*Activity-triplet*) représentant la demande du département des finances dans ARBAS. L'action *Design-1* est déclenchée par le département des finances et est utilisée pour représenter de manière formelle le dialogue entre ce département et l'équipe des développeurs. Comme l'indique la Figure II.4., *Design-1* est supposée consommer les ressources de l'équipe des développeurs et fournir plus de fonctions supportées par le système au département des finances.

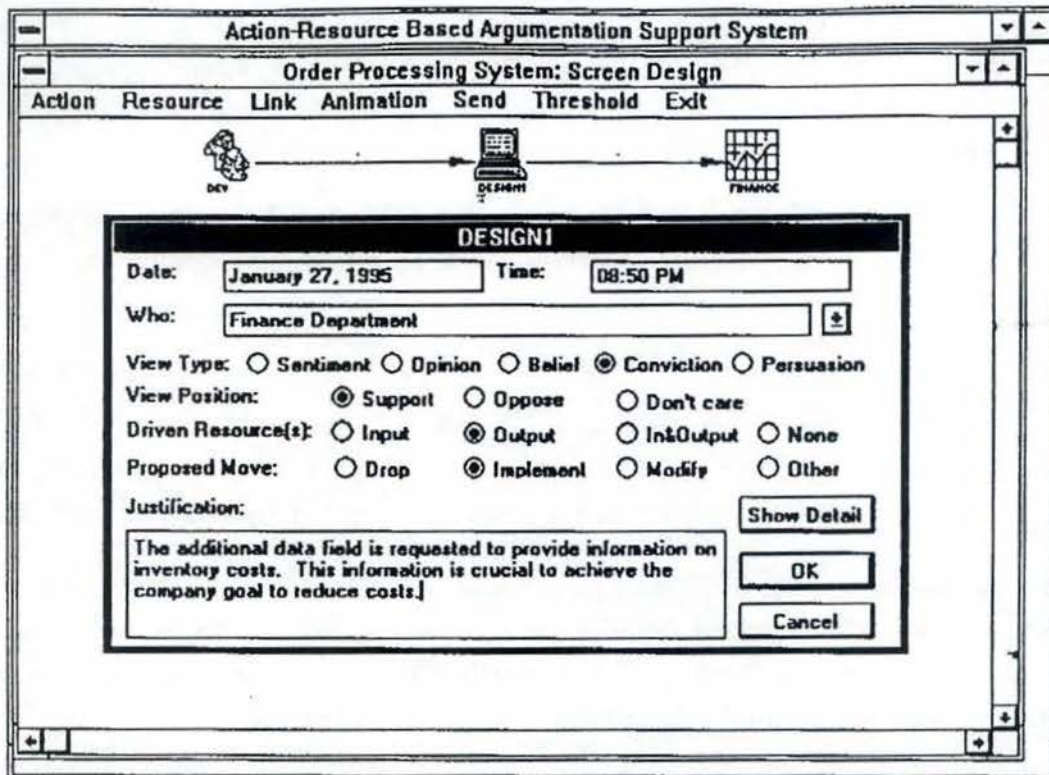


Figure II.4. : Les exigences du département des finances - Adapté de Bui et al. (1995)

Deux *action-objects* sont attachés à l'activité : un objet pour soutenir l'argumentation qui est constable par le bouton *Show Detail* et un langage pour supporter l'argumentation (la fenêtre *Design-1*).

En réaction à cette première proposition d'action, le département des ventes souhaite aussi que certains changements soient apportés au système (*Design-2* dans la Figure II.5.).



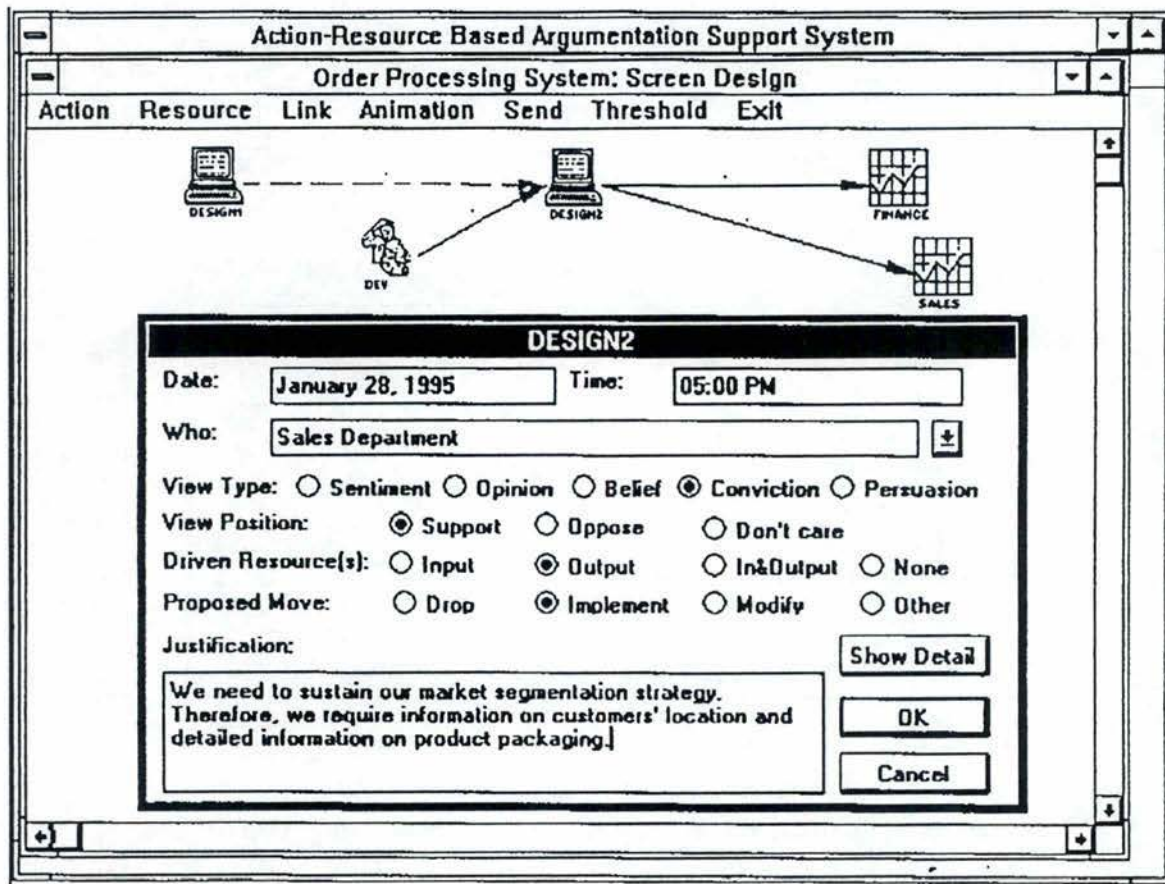


Figure II.5. La réaction du département des ventes - Adapté de Bui et al. (1995)

Malheureusement, l'équipe des développeurs n'a pas les moyens (temps, ressources) pour faire face à ces deux demandes et doit donc s'y opposer. Le département des ventes renonce à une partie de ses exigences concernant le logiciel à condition de recevoir davantage de ressources (*Design-3* dans Figure II.6.) . La Direction demande au département de production de postposer ses exigences en matière de systèmes d'informations, ainsi l'équipe des développeurs peut consacrer plus de temps au système de commande. L'équipe informatique accepte la proposition *Design-3* ce qui entraîne la résolution du conflit.

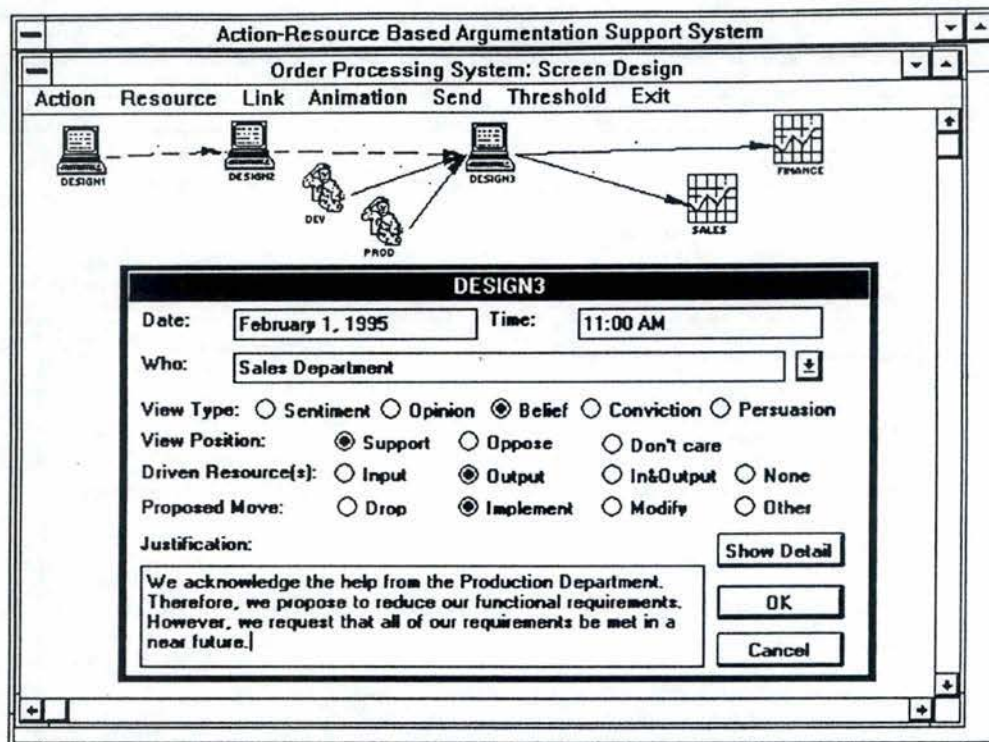


Figure II.6. La proposition du département des ventes - Adapté de Bui et al. (1995)

La représentation d'un problème dans ARBAS est basée sur la relation action-ressource. Ceci nécessite que les acteurs concernés par la résolution du problème identifient quelles actions sont à entreprendre pour aboutir à une solution et quelles ressources sont affectées par ces actions.

L'ensemble composé par une action et les éventuelles ressources affectées par cette action constitue une activité qui est représentée par un triplet :

Ressources en entrée | Action | Ressources en sortie.

L'évolution d'un problème est représentée par le chaînage de triplets d'activités au cours du temps (voir la chaîne *Design-1*, *Design-2*, *Design-3* dans Figure II.6.). Ceci permet, par exemple, de retracer toute la dynamique d'un processus de négociation. Les acteurs doivent pouvoir s'exprimer à propos d'une action proposée et le moyen d'expression qui leur est offert est la "vue" (voir la fenêtre intitulée *Design-3* dans Figure II.6. qui représente la vue posée par le département des ventes sur l'action *Design-3*). Une vue peut représenter une opinion, un sentiment, une croyance, une conviction ou une persua-



sion. Il est, dès lors, possible de savoir qui a proposé telle action et avec quelle force, quelle a été la (ré-)action proposée face à une action, qui supporte, s'oppose ou encore ne se sent pas concerné par telle action.

#### **II.4.6. Les limites et les extensions d'ARBAS**

Nous présentons dans cette partie les limites d'ARBAS en tant que système d'aide à la négociation et nous évoquons les modifications que nous avons apportées à ARBAS pour faire face à ces différentes limites.

Du point de vue de la négociation, les limites d'ARBAS trouvent leur origine dans la représentation d'un problème sous la forme d'une chaîne d'actions et dans l'utilisation de vues comme moyen d'expression des acteurs.

##### **Critiques du chaînage des actions**

Dans ARBAS, l'évolution d'un problème est représentée par une chaîne d'activités. Ce mode de représentation s'applique particulièrement bien à la négociation bilatérale puisque tour à tour les deux parties peuvent émettre une nouvelle proposition. Une partie émet une proposition et attend la réaction (nouvelle proposition) de la partie adverse. Une fois que la partie adverse a réagi en émettant une nouvelle proposition, la première partie réagit à son tour, et ainsi de suite. Malheureusement, si le chaînage des activités convient bien aux négociations bilatérales, nous ne pouvons en dire autant pour les négociations impliquant plus de deux parties. En effet, lorsque plusieurs propositions sont émises en réaction à la dernière action proposée, il faut trouver un système qui sélectionne la réaction qui sera retenue parmi toutes celles proposées. Pour le choix de cette réaction, une règle de sélection du style FIFO (First-In, First-Out) peut être envisagée, mais avec un tel mécanisme seule la première réaction est retenue et les autres seront perdues alors qu'elles avaient peut-être plus de chances d'aboutir à la solution. Le vote peut aussi constituer un autre mécanisme de sélection. Seulement, étant donné que les propositions sont émises par des acteurs en conflit, on risque d'arriver rapidement à des situations de blocage étant donné que chacun voudra défendre sa proposition. La situation peut devenir rapidement absurde si l'intervention d'un système d'aide à la négociation est nécessaire pour résoudre ce problème de sélection !

Dans ARBAS, il ne peut y avoir qu'une seule réaction (nouvelle proposition d'action) à une proposition d'action et une proposition d'action ne peut être émise que comme réaction à la dernière proposition. Nous énoncerons donc la limite suivante :

**Limite 1 : ARBAS représente un problème par un chaînage simple des activités proposées.**

Les conséquences de cette limite sont multiples.

**Conséquence 1 : ARBAS ne permet pas d'effectuer des tours de table.**

En effet, nous définirons le tour de table comme étant la possibilité offerte à tous les acteurs d'émettre, à un moment donné, une proposition d'action exprimant leur sentiment sur la façon de résoudre le problème. Mais, étant donné qu'ARBAS ne permet pas d'avoir plusieurs actions à un même moment, ARBAS ne peut pas mémoriser les tours de table.

**Conséquence 2 : ARBAS ne permet pas à un acteur d'émettre plusieurs propositions en même temps.**

Cette deuxième conséquence peut être considérée comme un cas particulier de la première où les propositions ne sont plus présentées par tous les acteurs mais par un seul d'entre-eux. Si un acteur désire proposer plusieurs actions, il ne peut le faire qu'en les englobant dans une grosse proposition, les actions proposées étant alors séparées par des OU. Par exemple, si nous reprenons l'exemple de la mise à jour du système de commande de produits, la première proposition du département des finances (*Design-1*) aurait pu être "On veut que le système informatique soit modifié OU on veut acheter un nouveau système informatique". Malheureusement, dans ces conditions, il est difficile pour les autres acteurs de critiquer de manière précise les différentes parties de la proposition. La Direction pourrait, par exemple, supporter la première partie de la proposition tout en rejetant la deuxième. Les deux critiques de la Direction se retrouvent incluses dans la justification de la vue posée par la Direction sur l'action proposée (*Design-1*) et s'il y a des objets (ex : un bon de commande) supportant la critique, on ne sait pas à quelle partie de la critique ces objets se rapportent.

**Conséquence 3 : ARBAS peut forcer les acteurs à travailler sur une proposition inintéressante.**

Etant donné que les acteurs doivent toujours repartir de la dernière proposition d'action, la qualité de celle-ci est importante. Il faut que cette dernière proposition d'action fasse avancer



le problème vers la solution. Le problème d'ARBAS est que, même si la dernière action proposée est ridicule ou tout à fait hors de propos, les acteurs doivent s'en servir comme point de départ pour les nouvelles propositions d'action. Dans ARBAS, un acteur pourrait émettre intentionnellement des propositions ridicules de manière à bloquer la négociation. Cette situation doit pouvoir être évitée.

**Conséquence 4 : ARBAS permet d'éliminer la dernière proposition trop facilement.**

En effet, il suffit qu'un acteur décide d'émettre une nouvelle proposition pour que l'ancienne ne soit plus prise en considération. Les acteurs doivent, dès lors, oublier la proposition sur laquelle ils étaient concentrés pour ne plus s'intéresser qu'à la nouvelle et ce, quelque soit la qualité de celle-ci (cf. conséquence 3).

**Conséquence 5 : ARBAS n'offre pas une vue d'ensemble sur le problème.**

En effet, avec la représentation chaînée des activités, les regards se concentrent toujours sur la dernière proposition et il n'y a pas de vue d'ensemble sur le problème. A un moment donné, il est difficile de savoir quelle est la situation exacte dans la résolution du problème, quelles sont les propositions de tous les acteurs. Pour les connaître, il faut passer en revue toutes les actions proposées et les analyser.

Pour résoudre ces différents problèmes liés au chaînage d'actions proposées, nous proposons de ne plus travailler avec des séquences de propositions d'actions mais avec des séquences d'états où un état est composé d'un ensemble de propositions.

**L'état d'une négociation** (à un moment donné) est la situation générale dans laquelle se retrouve la négociation. Cette situation est décrite par les caractéristiques du problème (titre, description, acteurs impliqués) et par l'ensemble des états des propositions d'actions émises par un ou plusieurs acteurs et qui coexistent à ce moment donné.

**L'état d'une proposition** à un moment donné est la situation dans laquelle se retrouve la proposition à ce moment donné. Cette situation est décrite par le contenu de la proposition (titre, action menée par cette proposition) et par les différentes attaques ou supports posés sur la proposition.

Etant donné que le chaînage d'états permet l'émission de plusieurs propositions en même temps, nous proposons d'introduire un protocole de négociation régissant les règles d'intervention des différents acteurs.

Le **protocole** d'une négociation est la méthode selon laquelle les acteurs peuvent intervenir au cours de cette négociation.

Nous proposons d'introduire deux types de protocole dans ARBAS : l'intervention libre et le **tour de table**. Lorsque le protocole défini est l'intervention libre, chaque acteur peut intervenir en émettant une nouvelle proposition (ou en en modifiant une) à n'importe quel moment. Chaque acteur travaille donc en connaissant l'ensemble des propositions antérieures. Pour le protocole du tour de table, chaque acteur est invité à émettre une nouvelle proposition à un moment précis (e.g. toutes les semaines). Dans la pratique, il s'agira d'attendre que chacun ait émis sa proposition avant de présenter le nouvel état de la négociation. Le tour de table est un protocole intéressant car il oblige les acteurs à émettre des propositions et permet d'augmenter la participation à la résolution du problème. Puisque le tour de table force les acteurs à émettre des propositions à un moment donné, la recherche d'une solution peut être plus rapide. Ce protocole permet aussi de savoir qui est intéressé par la négociation et d'éliminer les propositions qui n'ont plus de raison d'être, vu le manque d'intérêt porté par leurs auteurs. L'inconvénient d'un tour de table est que les propositions risquent de manquer d'intérêt car comme les acteurs ne connaissent pas les propositions des autres parties, ils ne veulent pas céder trop de choses sans savoir à l'avance ce qu'ils auront en retour. Cet inconvénient ne se rencontre pas dans le cas de l'intervention libre. Néanmoins, le défaut de l'intervention libre est qu'il faut parfois attendre longtemps avant que quelqu'un n'émette une nouvelle proposition. En bref, le choix d'un protocole de négociation n'est pas facile et dépend du type de problème à résoudre. Remarquons que le style choisi peut être un mélange des deux protocoles décrits. Il doit donc pouvoir être modifié par le secrétaire en cours de négociation.

Selon Schwabe (1994), il faut faire un meilleur usage des technologies de l'information pour améliorer la confidentialité et ce, afin de permettre des discussions plus libres ou pour protéger les intérêts de l'individu. Nous proposons donc d'instaurer un protocole de négociation dans lequel les acteurs peuvent s'exprimer de manière anonyme. ARBAS assure l'anonymat des acteurs par l'utilisation de pseudonymes.

Etant donné que nous avons défini trois niveaux d'accès à ARBAS (public, autorisé et privé), nous avons aussi ajouté les attributs `login_name` et `mot de passe` aux acteurs.



## Critiques des vues

L'utilisation de vues comme moyen d'expression des acteurs est aussi à l'origine d'un certain nombre de problèmes. En effet, avec les vues telles qu'elles ont été définies dans ARBAS, chaque acteur ne peut poser qu'un seul regard sur une proposition d'action. Généralement, une proposition est accompagnée d'arguments et ceux-ci doivent pouvoir être critiqués par les acteurs. Les vues d'ARBAS mélangent trop de concepts différents tels que la proposition d'une action, la critique d'une proposition d'action et l'émission d'arguments. Lorsqu'un acteur désire proposer une action, il doit créer une vue dans laquelle *View\_Position* est *Support* et *Proposed\_Moved* est *Implement*. Pour critiquer, l'acteur doit poser une vue où *View\_Position* est *Support* ou *Oppose* et *Proposed\_Moved* est *Other*. Cette représentation manque de clarté et il nous semble plus judicieux de séparer les vues en deux catégories, à savoir celles qui concernent les propositions d'actions et celles relatives aux arguments posés sur ces propositions. Ainsi, toutes les informations propres à la proposition sont accessibles plus facilement. Il est possible de connaître l'acteur qui est à l'origine de la proposition, une description complète de l'action proposée sans devoir effectuer autant de manipulations que dans ARBAS ou pour connaître le créateur d'une proposition, par exemple, il faut passer en revue toutes les vues pour voir laquelle est la plus ancienne.

Nous ajouterons que, dans les vues, beaucoup d'arguments sont regroupés dans la justification ce qui ne facilite pas la critique de chacun d'eux. Il faut que les acteurs puissent émettre des contre-arguments pour chaque argument proposé. Pour résoudre ce problème, nous proposons de remplacer la justification par une liste d'arguments. Ainsi, les acteurs peuvent poser des contre-arguments pour chaque argument émis. Il est même possible d'émettre des contre-contre-arguments et ainsi de suite... Chaque argument, tout comme chaque proposition peut être référencé par un objet (e.g. une note de service) qui correspond au concept d'action-object présent dans la première version d'ARBAS.

Un **argument** est un support, une attaque ou un avis neutre qui a été posé sur une proposition ou sur un argument de manière à renforcer, détruire ou maintenir la validité de cette proposition ou de cet argument.

Nous énoncerons donc la limite suivante :

**Limite 2 : Les vues sont trop générales pour servir de moyen d'expression efficace des acteurs.**

Pour se rendre compte qu'un accord est intervenu dans un problème de négociation, il faut analyser toutes les vues posées sur la dernière proposition. Si elles sont toutes favo-

rables à cette proposition (*View\_Position* est *Support* et *Proposed\_Moved* est *Implement*), c'est qu'un accord est intervenu. De manière à faciliter cette analyse, nous proposons d'ajouter le concept d'accord à ARBAS. Un problème de négociation peut donner lieu à un accord. Celui-ci représente la mise en oeuvre d'une proposition soutenue par les acteurs impliqués dans la négociation.

Un accord est un consensus auquel sont éventuellement arrivés les acteurs de la négociation.

### Critique du concept de ressource

Comme nous l'avons rappelé plus haut, ARBAS est un langage d'argumentation basé sur la relation action-ressource. Cette notion de ressource est intéressante mais difficilement évaluable. Généralement, pour une action, il est difficile d'estimer la quantité de ressources qu'elle consomme et qu'elle produit. Dans le problème "Que fait-on ce soir ?", si un acteur propose d'aller au cinéma, cette action consomme des ressources (temps, argent) estimables et produit des ressources (joie, par exemple) non calculables. Dès lors, nous suggérons de ne plus prendre en considération les ressources dans la nouvelle version d'ARBAS. De plus, les ressources sont gérées par le niveau de planification et de prises de décision et, il nous semble, dès lors, que la présence de ce concept serait plus judicieuse au deuxième niveau de la pyramide (Figure II.1.)

En résumé, différents changements ont été apportés à ARBAS de manière à mieux répondre à ses limites au niveau de sa représentation du problème et au niveau du moyen d'expression offert aux acteurs (vues) :

- Remplacement du chaînage des actions proposées par un chaînage des états du problème
- Introduction d'un protocole de négociation
- Possibilité d'anonymat
- Remplacement des vues par des arguments
- Suppression du concept de ressource
- Ajout du concept d'accord



#### II.4.7. ARBAS étendu dans le cadre de la négociation

Au point précédent, nous avons critiqué ARBAS et nous y avons apporté quelques modifications. Nous présentons, ci-après (Figure II.7.), un nouveau schéma graphique représentant la première étape de l'extension d'ARBAS.

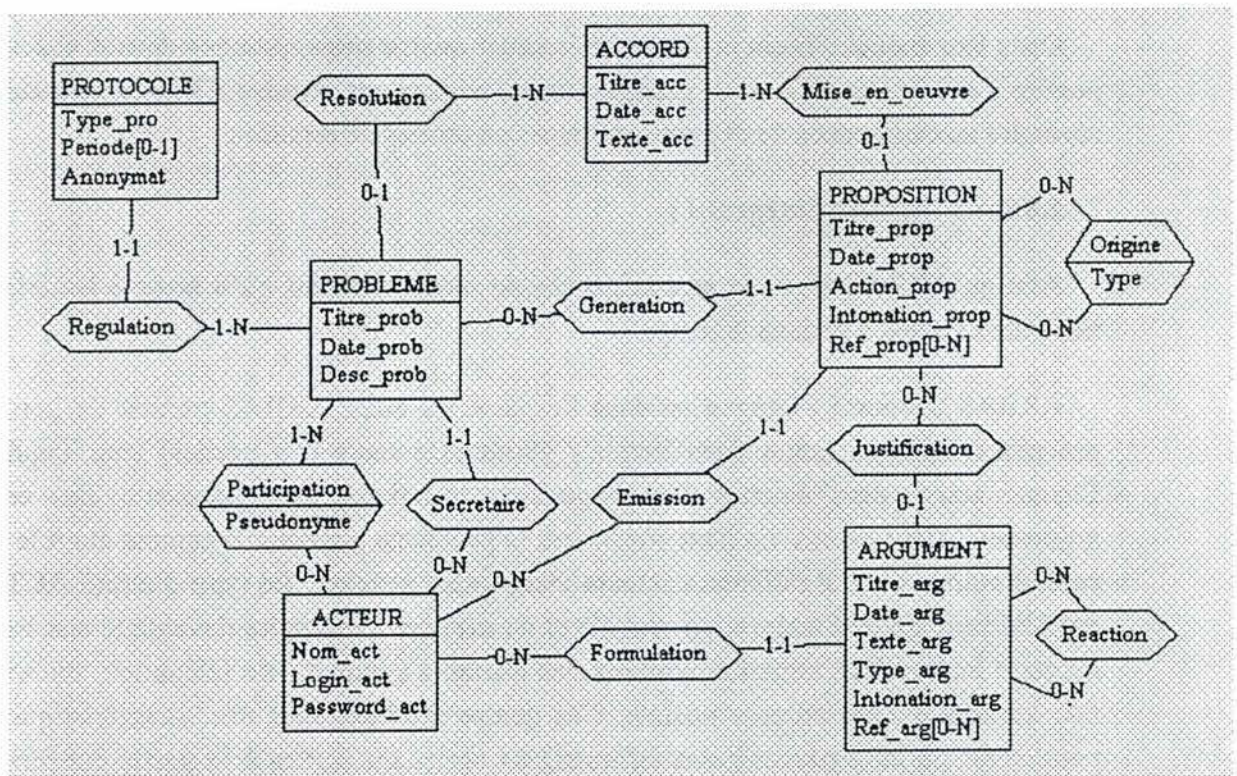


Figure 11.7. : Le schéma graphique d'ARBAS étendu dans le cadre de la négociation

## 11.5. ARBAS96 en tant que SISMO

Dans cette section, nous montrons de quelle manière nous avons étendu ARBAS pour qu'il réponde davantage aux exigences des Systèmes d'Informations Supportant de la Mémoire Organisationnelle (SISMOs). Le point de référence de cette extension est le modèle de Stein et Zwass, présenté dans la première partie (I.7.2.), et envisagé comme un cadre de travail incluant un ensemble de SISMOs possibles. Par hypothèse de travail, nous considérons le modèle tel qu'il est décrit par ses auteurs, et nous nous en servons comme modèle de référence pour l'extension d'ARBAS.



Afin qu'ARBAS puisse répondre à un maximum des exigences présentées dans le modèle, il a fallu l'étendre selon deux directions. La première direction est une extension au niveau des concepts pris en considération par ARBAS. En introduisant de nouveaux concepts, ARBAS peut mieux supporter la mémoire organisationnelle. La seconde direction est une extension au niveau des requêtes offertes par ARBAS.

### ***II.5.1. Extension d'ARBAS en termes de concepts***

Dans leur modèle, Stein et Zwass présentent quatre sous-systèmes devant chacun répondre à des besoins. Dans un premier temps, nous allons étendre ARBAS de manière à répondre aux exigences de ces sous-systèmes en termes de données.

#### **Le Sous-Système Intégratif :**

Afin qu'ARBAS réponde aux exigences du modèle, il faut qu'il intègre les informations dans l'espace et dans le temps.

ARBAS supporte l'intégration dans l'espace. En effet, ARBAS permet de gérer des processus de négociation dans lesquels plusieurs acteurs (ou parties) sont impliqués. Chaque acteur peut émettre des propositions d'actions, des arguments, des contre-arguments. Dans ce cas, on peut considérer le problème à résoudre comme étant le point central autour duquel différents acteurs viennent se pencher. Le type d'entité ACTEUR est l'élément clé de l'intégration spatiale et de manière à répondre davantage aux besoins de cette intégration, nous lui ajoutons des attributs : numéro de téléphone, numéro de fax, adresse e-mail, adresse postale. Ces quatre attributs sont des caractéristiques qui permettent de faire parvenir des informations orales ou écrites à la partie concernée et constituent des informations sur l'intégration spatiale de l'acteur. Nous étendons aussi le schéma par un nouveau type d'entité DEPARTEMENT attaché au type d'entité ACTEUR par une relation ASSIGNATION. Cette extension permet de mémoriser l'activité d'un acteur au cours du temps. A telle époque, il occupait tel poste dans tel département.

**Un département** est une section de l'organisation à laquelle l'acteur est attaché.

Il nous faut tout de même remarquer que l'intégration dans l'espace ne peut se faire que si ARBAS est accompagné d'une technologie permettant de distribuer les informations entre les utilisateurs du système (e.g. messagerie électronique).



Pour l'intégration temporelle, ARBAS a été étendu par des attributs temporels (date, heure) dans les types d'entités PROBLEME, PROPOSITION, ARGUMENT, et dans les types d'entités ASSIGNATION, PARTICIPATION, SECRETAIRE. Ces extensions sont particulièrement utiles car elles permettent de mieux comprendre le contexte dans lequel une proposition, par exemple, a été émise.

### **Le Sous-Système Adaptatif :**

Lorsqu'un problème surgit, il est possible qu'il trouve son origine dans un ou plusieurs événements. Ces événements peuvent être internes ou externes à l'organisation. La résolution du problème montre de quelle manière l'organisation réagit face à ces événements, autrement dit, comment l'organisation s'adapte à des événements qui se sont produits dans l'organisation ou dans son environnement. Nous avons rajouté le concept d'événement à ARBAS. Remarquons toutefois qu'ARBAS96 ne s'occupera pas de la gestion des événements (capture, filtrage, organisation) mais offrira la possibilité aux acteurs de s'en servir comme références dans leurs propositions, arguments et buts. En limite, seuls les événements qui ont trait à une négociation sont retenus dans ARBAS96.

Un **événement** est un fait qui s'est produit à un moment donné dans l'organisation ou dans son environnement.

Les informations relatives à l'environnement sont aussi présentes dans les références des types d'entités PROPOSITION, ARGUMENT, BUT.

Une **référence** est un objet sur lequel une proposition, un argument ou un but peut se référer de manière à augmenter leur qualité.

Le concept de référence est identique à la notion d'*action\_object* présente dans la première version d'ARBAS. Les acteurs disposent d'informations sur l'environnement ce qui leur permet de mieux argumenter leurs propositions et donc d'augmenter la qualité de ces propositions.

### **Le Sous-Système de Réalisation de buts :**

Pour qu'ARBAS puisse identifier, stocker, modifier et évaluer des buts, nous avons étendu le schéma Entité-Association d'ARBAS par les types d'entités BUT, CRITERE et OBJECTIF ORGANISATIONNEL.

Un but est un objectif que désire atteindre un acteur dans le cadre de la résolution du problème.

Le type d'entité CRITERE permet d'évaluer le degré de réalisation des buts et répond de ce fait à un des besoins exprimés dans le sous-système.

Un critère est un élément qui permet d'évaluer dans quelle mesure un but est ou n'est pas atteint.

Remarquons que l'introduction de ces deux types d'entités permet aussi d'améliorer la phase de définition du problème et la phase de recherche de la solution présentes dans les processus de négociation (Bui et Shakun (1995)).

Afin qu'ARBAS fournisse des informations à propos des objectifs organisationnels, nous avons introduit le type d'entité OBJECTIF ORGANISATIONNEL que nous avons rattaché au type d'entité BUT. Les buts des acteurs peuvent s'appuyer sur, ou contribuer à, des objectifs de l'organisation. De la sorte, ARBAS mémorise les objectifs organisationnels et offre un lien entre ces objectifs et les problèmes de négociation qui ont été mémorisés.

Un objectif organisationnel est un but visé par l'organisation au niveau stratégique.

#### **Le Sous-Système de Maintenance des procédures de travail :**

Avec les données présentes dans ARBAS, il est possible de mémoriser une partie de l'historique de travail des individus et de percevoir le climat général dans lequel baigne l'organisation.

ARBAS mémorise des résolutions de problèmes impliquant des acteurs et il est donc possible de retrouver tous les problèmes auxquels un acteur a participé. Ces problèmes permettent d'inférer les caractéristiques des acteurs. La qualité des propositions et des critiques d'un acteur reflète les capacités de celui-ci. Les intonations posées sur les propositions permettent de saisir le caractère de l'individu. Quelqu'un dont les propositions sont toujours accompagnées d'une intonation forte (e.g. Persuasion) pourra être considéré comme une personne décidée. Par contre, lorsqu'un acteur ne pose que des propositions à faible degré d'intonation, il pourra être considéré comme peu sûr de lui.

L'évolution des buts d'un acteur reflète aussi son degré de flexibilité. Si un acteur ne modifie jamais ses buts, il est très difficile de le faire changer de position.



L'analyse des problèmes peut aussi montrer dans quelle mesure le climat de l'organisation est bon ou mauvais. Par exemple, si parmi les problèmes enregistrés, un grand nombre concerne des conflits de personnes, alors le climat général pourra être considéré comme mauvais. Par contre, si très peu de conflits relatifs au personnel de l'organisation sont mémorisés alors le climat pourra être considéré comme étant favorable. Le fait d'analyser les processus de résolution des conflits indique aussi dans quelle mesure les acteurs sont motivés ou pas. En effet, si les problèmes sont résolus relativement vite cela indique que le personnel est particulièrement motivé, dynamique. Par contre, si pour un problème d'envergure moyenne, il faut attendre beaucoup de temps avant d'atteindre une solution car personne n'émet de proposition ou parce que personne ne veut changer sa position, le climat général pourra être considéré comme mou, inactif.

Une autre manière d'évaluer le travail d'un acteur est de lui offrir la possibilité de s'exprimer par rapport aux problèmes auxquels il a participé. Il peut émettre une appréciation par rapport à sa propre attitude ou par rapport au déroulement général d'une négociation. De plus, il permet d'enregistrer les conséquences positives ou négatives de l'accord. Nous ajoutons un type d'entité FEED-BACK au schéma Entité-Association.

Un <b>feed-back</b> est une analyse rétroactive portée sur un accord.
-----------------------------------------------------------------------

Il est important de souligner qu'une donnée peut jouer plusieurs rôles dans les sous-systèmes proposés. Par exemple, un feed-back est un composant qui est aussi important pour le sous-système adaptatif étant donné qu'il permet aux acteurs de disposer d'informations à posteriori sur la qualité des accords réalisés. De la sorte, ces informations permettent aux participants d'adapter leur comportement présent aux situations passées.



## II.5.2. Schéma Entité-Association d'ARBAS96

### Le schéma graphique

La Figure II.8. représente le schéma graphique d'ARBAS96. Il correspond à l'extension d'ARBAS dans le cadre de la mémoire organisationnelle. Le modèle Entité-Association est un modèle qui permet d'exprimer la sémantique des données *mémorisables* et/ou *véhiculables* à l'aide des concepts d'entité, d'association, d'attributs et du mécanisme des contraintes d'intégrité (Bodart et Pigneur (1989)).

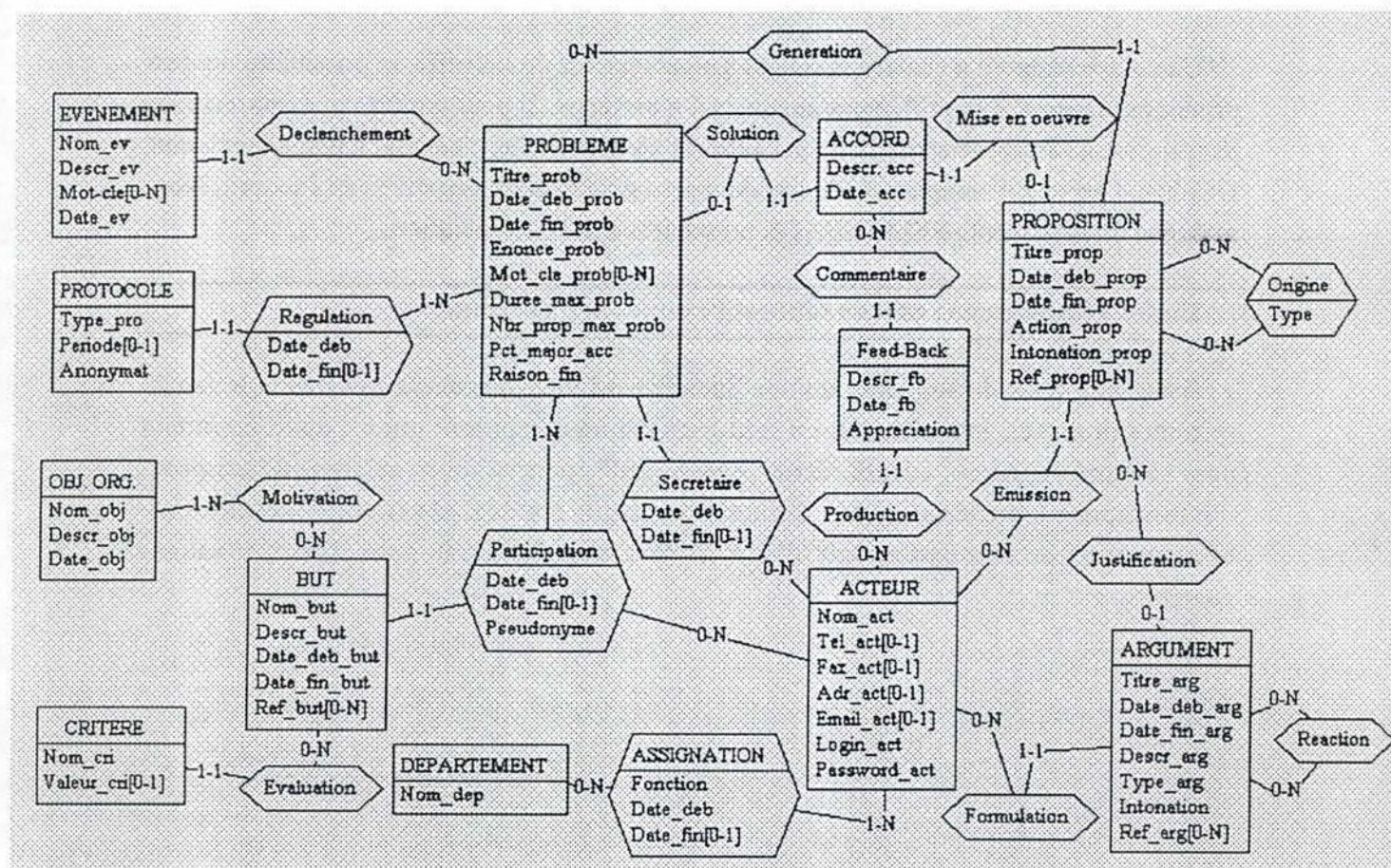


Figure II.8. : Le schéma graphique d'ARBAS étendu à la mémoire organisationnelle



## Les définitions

### Les Types d'entité

#### Evénement

**Définition :** Un événement est un fait qui s'est produit à un moment donné dans l'organisation ou dans son environnement et qui est à l'origine de la négociation.

**Attributs :**

Nom\_ev : Nom associé à un événement.  
Descr\_ev: Description de l'événement.  
Mot-clé: Mots caractéristiques associés à l'événement.  
Date\_ev: Moment de l'apparition de l'événement.

**Identifiant :** Nom\_ev.

**Exemple :**

Nom\_ev : 'Nouveau concurrent'.  
Descr\_ev: 'La société Video fait son entrée sur le marché de l'audiovisuel'.  
Mot-clé: {'Audiovisuel', 'Pénétration de Marché'}.  
Date\_ev: '28/10/95 - 8.40'.

#### Problème

**Définition :** Un problème est la description du sujet d'une négociation.

**Attributs :**

Titre\_prob: Nom associé au problème pour permettre de l'identifier.  
Date\_deb\_prob: Date de création du problème.  
Date\_fin\_prob: Date de terminaison du problème.  
Enoncé\_prob: Enoncé complet du problème de la négociation.  
Mot\_cle\_prob: Mots caractéristiques associés au problème.  
Durée\_max\_prob: Temps accordé à la résolution du problème.  
Nbr\_prop\_max: Nombre maximum de propositions que l'on peut émettre pour un problème.  
Pct\_major\_acc: Pourcentage à atteindre pour valider un accord.

Raison-fin: Expression des conditions de terminaison de la négociation.

Identifiant : Titre\_prob

Exemple :

Titre\_prob: 'Softwood Lumber'.  
Date\_deb\_prob: '19/05/86 - 9:00'.  
Date\_fin\_prob: '31/12/86 - 23:55'.  
Enoncé\_prob: 'Le problème Softwood Lumber est lié à la forte position du Canada sur le marché du bois américain'.  
Mot\_cle\_prob: {'Commerce', 'International', 'Bois'}.  
Durée\_max\_prob: '6 mois'.  
Nbr\_prop\_max: '20'.  
Pct\_major\_acc: '100 %'.  
Raison-fin: 'accord'.

## Acteur

Définition : Un acteur est une personne ou une organisation qui peut intervenir dans un problème de négociation.

Attributs :

Nom\_act : Libellé du nom de l'acteur.  
Tel\_act : Numéro de téléphone de l'acteur.  
Fax\_act : Numéro de fax de l'acteur.  
Adr\_act : Adresse de l'acteur.  
Email\_act : Adresse électronique de l'acteur.  
Login\_act : Identification de l'acteur pour accéder aux négociations.  
Password\_act: Authentification de l'acteur.

Identifiant : Nom\_acteur

Exemple :

Nom\_act : 'Gérard Menvuça'.  
Tel\_act : '081/22.16.74'.  
Fax\_act : '081/22.16.75'.  
Adr\_act : '12, avenue des myopes - 5000 Bouge'.



Email\_act : 'gme@info.fundp.ac.be'.  
Login\_act : 'gmenvuca'.  
Password\_act: 'kfr425r7'.

### **Objectif organisationnel**

Définition : But visé par l'organisation au niveau stratégique.

Attributs :

Nom\_obj : Nom associé à l'objectif organisationnel pour permettre de l'identifier.  
Descr\_obj : Description de l'objectif organisationnel.  
Date\_obj : Date de création de l'objectif organisationnel.

Identifiant : Nom\_obj

Exemple :

Nom\_obj : 'Réduction coûts'.  
Descr\_obj : 'Réduire tous les coûts de 20 % avant la fin de l'année'.  
Date\_obj : '20/10/95'.

### **But**

Définition : Un but est un objectif que désire atteindre un acteur dans le cadre de la résolution du problème.

Attributs :

Nom\_but : Nom associé au but pour permettre de l'identifier.  
Descr\_but : Enoncé explicatif du but.  
Date\_deb\_but : Date de création du but.  
Date\_fin\_but: Date de suppression du but.  
Réf\_but: « Objet » sur lequel le but se réfère.

Identifiant : Descr\_but.

Exemple :

Nom\_but : 'Équité du marché'  
Descr\_but : 'Pouvoir arriver à l'équité du marché'.

Date\_deb\_but : '20/12/95'.  
Date\_fin\_but: '25/01/96'.  
Réf\_but: Lien vers un rapport enregistré dans un fichier.

### **Critère**

Définition : Un critère est une condition à satisfaire pour atteindre un but.

Attributs :

Nom\_cri : Nom identifiant le critère.  
Valeur\_cri : Quantification du critère.

Identifiant : Nom\_cri

Exemple :

Nom\_cri : 'Bénéfice minimal'.  
Valeur\_cri : '200.000'.

### **Proposition**

Définition : Une proposition est une action proposée par un acteur pour résoudre le problème de négociation.

Attributs :

Titre\_prop : Nom identifiant la proposition.  
Date\_deb\_prop : Date de création de la proposition.  
Date\_fin\_prop : Date de suppression de la proposition.  
Action\_prop : Enoncé complet de l'action proposée.  
Intonation\_prop : Représentation de la conviction de l'acteur envers sa proposition. { Sentiment, Opinion, Croyance, Conviction, Persuasion }.  
Réf\_prop : « Objet » sur lequel la proposition se réfère.

Exemple :

Titre\_prop : 'Abattre bétail'.  
Date\_deb\_prop : '20/04/96 - 9:00'.



Date\_fin\_prop : '20/04/96 - 12:00'.  
 Action\_prop : 'Pour enrayer la maladie de la vache folle, il faut abattre le bétail'.  
 Intonation\_prop : 'Conviction'.  
 Réf\_prop : lien OLE vers un rapport d'un vétérinaire.

Identifiant : Titre\_prop

### Argument

Définition : Un argument est un support, une attaque ou un avis neutre qui a été posé sur une proposition ou sur un argument de manière à renforcer, détruire ou maintenir la validité de cette proposition ou de cet argument.

Attributs :

Titre\_arg : Nom identifiant l'argument.  
 Date\_deb\_arg : Date de création de l'argument.  
 Date\_fin\_arg : Date de suppression de l'argument.  
 Descr\_arg : Description de l'argument.  
 Intonation : Représentation de la conviction de l'acteur envers son argument. { Sentiment, Opinion, Croyance, Conviction, Persuasion }.  
 Réf\_arg : « Objet » sur lequel l'argument se réfère.

Identifiant : Titre\_arg

Exemple :

Titre\_arg : 'Transmissible'.  
 Date\_deb\_arg : '20/04/96 - 9:00'.  
 Date\_fin\_arg : '20/04/96 - 10:00'.  
 Descr\_arg : 'La maladie de la vache folle est transmissible à l'homme'.  
 Type\_arg : 'Support'.  
 Intonation : 'Croyance'.  
 Réf\_arg : lien vers un rapport d'un médecin.

### **Protocole**

**Définition :** Méthode selon laquelle les acteurs vont pouvoir intervenir dans la négociation.

**Attributs :**

**Type\_pro :** {Tour de table, Intervention libre, Panaché}.  
**Période :** Intervalle de temps séparant deux tours de table successifs.  
**Anonymat:** Booléen indiquant si les acteurs interviennent de manière anonyme ou pas.

**Identifiant :** Type\_pro

**Exemple :**

**Type\_pro :** 'Intervention libre'.  
**Période :** Vide dans le cas d'un protocole à intervention libre.  
**Anonymat:** 'Vrai'.

### **Département**

**Définition :** Section de l'organisation à laquelle l'acteur est attaché.

**Attributs :**

**Nom\_dép :** Libellé du nom du département.

**Identifiant :** Nom\_dép

**Exemple :**

**Nom\_dép :** 'Finances'.

### **Accord**

**Définition :** Consensus auquel sont arrivés les acteurs de la négociation.



Attributs :

Descr\_acc : Description de l'accord.

Date\_acc : Date à laquelle a été conclu l'accord.

Identifiant : Rôle 'a pour solution' joué par problème

Exemple :

Descr\_acc : 'La France s'engage à importer de la viande anglaise si la Grande Bretagne élimine son problème de vache folle'.

Date\_acc : '20/06/96'.

### **Feed-back**

Définition : Analyse rétroactive portée sur un accord.

Attributs :

Descr\_fb: Description du feed-back.

Date\_fb : Date à laquelle a été porté la feed-back.

Appréciation : {Very Bad, Bad, Good, Very Good, Perfect}.

Identifiant : Rôle 'est commenté' joué par accord et rôle 'produit' joué par acteur

Exemple :

Descr\_fb: 'La mise en oeuvre de cet accord ne s'est pas avérée efficace car les anglais ont trouvé un moyen légal de détourner l'accord'.

Date\_fb : '20/07/96'.

Appréciation : 'Bad'.

### Types d'association

#### **Déclenchement**

Définition : Un déclenchement définit le ou les événements qui ont déclenché le problème de négociation.

Associe :            Événement, 1-1, déclenche.  
                         Problème, 0-N, est déclenché.

### **Participation**

Définition :        Une participation définit l'implication d'un acteur et de ses buts dans le cadre d'un problème de négociation.

Associe :            Problème, 1-N, a pour participant.  
                         Acteur, 0-N, participe.  
                         But, 0-1, justifie la participation.

Attributs :

    Date\_deb :      Date à laquelle un acteur entre en jeu dans un problème.  
    Date\_fin :       Date à laquelle un acteur quitte le problème de négociation.  
    Pseudonyme :   Surnom donné à un participant pour lui garantir l'anonymat.

Exemple :

    Date\_deb :      '20/10/95'.  
    Date\_fin :       '24/12/95'.  
    Pseudonyme :   'Nestor'.

### **Secrétaire**

Définition :        Un secrétaire est un acteur qui gère le bon déroulement d'un problème.

Associe :            Problème, 1-1, a pour secrétaire.  
                         Acteur, 0-N, est secrétaire.

Attributs :

    Date\_deb:       Date à laquelle un acteur devient le secrétaire du problème.  
    Date\_fin:        Date à laquelle un acteur n'assume plus sa fonction de secrétaire.

### **Motivation**



**Définition :** Une motivation définit le lien existant entre les objectifs organisationnels et les buts des acteurs émis dans le cadre d'un problème.

**Associe :** Objectif Organisationnel, 1-N, motive.  
But, 0-N, est motivé.

### **Evaluation**

**Définition :** Une évaluation définit un critère associé à un but.

**Associe :** Critère, 1-1, évalue.  
But, 0-N, est évalué.

### **Génération**

**Définition :** Une génération définit une proposition émise pour solutionner un problème.

**Associe :** Problème, 0-N, génère.  
Proposition, 1-1, est générée.

### **Production**

**Définition :** Une production définit l'émission d'un feed-back par un acteur.

**Associe :** Feed-back, 1-1, est produit.  
Acteur, 0-N, produit.

### **Emission**

**Définition :** Une émission définit une proposition qu'un acteur a formulée.

**Associe :** Acteur, 0-N, émet.  
Proposition, 1-1, est émise.

### **Origine**

Définition : Une origine définit quelle proposition provient d'une autre proposition.

Associe : Proposition, 0-N, a pour origine.  
Proposition, 0-N, est l'origine.

Attributs :  
Type: {Support, Oppose} définit le genre de relation entre deux propositions.

### **Justification**

Définition : Une justification définit un argument supportant une proposition.

Associe : Argument, 0-1, justifie.  
Proposition, 0-N, est justifiée.

### **Formulation**

Définition : Une formulation définit l'émission d'un argument par un acteur.

Associe : Argument, 1-1, est formulé.  
Acteur, 0-N, formule.

### **Réaction**

Définition : Une réaction définit l'argument d'un acteur vis-à-vis d'un autre argument.

Associe : Argument, 0-1, attaque.  
Argument, 0-N, est attaqué.

### **Assignment**

Définition : Une assignation définit l'occupation d'un acteur dans un département.

Associe : Acteur, 1-N, est assigné.



Département, 0-N, assigne.

**Attributs :**

- Fonction : Poste occupé par l'acteur au sein de son département.
- Date-deb : Date à laquelle l'acteur entre en fonction.
- Date-fin : Date à laquelle l'acteur quitte sa fonction pour éventuellement en occuper une autre.

**Exemple :**

- Fonction : 'Directeur commercial'.

**Régulation**

**Définition :** Une régulation définit le protocole de négociation appliqué à un problème.

**Associe :** Protocole, 0-1, régule.  
Problème, 1-N, est régulé.

**Attributs :**

- Date\_deb : Définit la date à laquelle le protocole a été appliqué.
- Date\_fin : Date à laquelle le protocole est modifié.

**Solution**

**Définition :** Une solution détermine si un accord est intervenu pour un problème.

**Associe :** Problème, 0-1, a pour solution.  
Accord, 1-1, solutionne.

**Mise en oeuvre**

**Définition :** Une mise en oeuvre définit la proposition qui a permis d'atteindre l'accord.

**Associe :** Proposition, 0-1, est mise en oeuvre.

Accord, 1-1, met en oeuvre.

### Commentaire

Définition : Un commentaire définit une évaluation par rapport à un accord.

Associe : Feed-Back, 1-1, commente.  
Accord, 0-N, est commenté.

### II.5.3. ARBAS96 vs ARBAS

Nous avons apporté différents changements dans ARBAS. Nous établissons dans Tableau II.3. la correspondance entre les deux versions d'ARBAS.

ARBAS	ARBAS96
Action_name	Proposition
Action_object	Référence dans Proposition et Argument et But
Activity triplet	-
Proposed_moved	-
Resource_name	-
Time stamp	Estampille sur les problèmes, propositions, arguments
View	Proposition et Argument
View_intonation	Intonation dans Proposition et Argument
View_justification	Action_prop ou Descr_Arg
View_owner	Auteur
View_position	Type d'argument ou de proposition

Tableau II.3. : Correspondance ARBAS / ARBAS96



#### **II.5.4. Extension d'ARBAS en termes de fonctionnalités**

Nous nous sommes, jusqu'à présent, intéressés à l'extension d'ARBAS en termes de concepts. Les différents concepts présents dans ARBAS96 représentent le contenu de la mémoire organisationnelle dans le cadre de la négociation.

Nous proposons maintenant d'étendre ARBAS au niveau de ses fonctions. Ces fonctions seront liées aux fonctions mnémoniques présentées dans les modèles de Walsh et Ungson (1991), Stein et Zwass (1995) et Morrison et Weiser (1996) : l'acquisition, la récupération et la maintenance des données.

##### **L'acquisition**

Morrison et Weiser (1996) distinguent dans l'acquisition trois types d'informations : les archives, les réponses et les insertions automatiques (voir point I.7.3.)

ARBAS96 est un système qui enregistre des informations de types archives. Lorsqu'un conflit a été perçu par l'organisation, une définition du problème proprement dit peut être énoncée et mémorisée par le système.

Remarquons que les caractéristiques du problème ne sont pas figées pour toute la négociation. Il est possible d'ajouter des acteurs en cours de négociation, de changer de secrétaire, etc... .

Tous les acteurs présents dans un problème doivent être mémorisés. Leurs propres caractéristiques (nom, adresse, téléphone, ...) sont stockées et un couple (*login*, *mot de passe*) leur est attribué pour leur permettre d'intervenir dans la négociation. Cette caractéristique rend le système plus sûr (voir II.8.3.). Dans le cas où la négociation est anonyme, les acteurs se voient attribuer un pseudonyme.

En début de négociation, les acteurs sont priés d'émettre leurs buts, les objectifs organisationnels sur lesquels se basent ces buts et les critères de satisfaction de ces buts. Au départ, ces buts seront peut-être évasifs, c'est pourquoi ARBAS permet de modifier les buts, objectifs et critères en cours de négociation.

Les acteurs vont intervenir dans la négociation en émettant des propositions et des arguments. Chacune de ces interventions doit être retenue dans la mémoire organisationnelle car elles représentent la base de la négociation.

Les propositions et les arguments vont évoluer au cours de la négociation. Toute modification est mémorisée pour qu'à tout moment il soit possible de comprendre l'évolution de la négociation.

Lorsque la négociation s'achève, ses conditions de terminaison sont acquises. La négociation peut avoir donné lieu à un accord, à un accord forcé (un certain pourcentage des acteurs soutiennent une proposition) ou à un abandon de la négociation suite à un dépassement du temps imparti ou à un nombre de propositions excessif.

Lorsqu'un accord survient, les acteurs de la négociation peuvent le critiquer en soumettant un feed-back. L'intérêt de cette critique est qu'elle est portée après coup. L'acteur n'est plus dans le cadre de la négociation. Il peut ainsi analyser la solution du problème sous un autre regard et en saisir les défauts et les qualités. L'acteur soumet, au système, son analyse sur la mise en oeuvre de la proposition qu'il accompagne d'une appréciation (*very bad, bad, good, very good, perfect*). Cette information est riche pour l'apprentissage.

En conclusion, l'acquisition, dans ARBAS96, a pour objectif de mémoriser toutes les informations relatives aux négociations. Cette acquisition est effectuée de manière volontaire par les participants ou le secrétaire d'une négociation.

### **La récupération**

La récupération d'informations peut être active (requêtes, filtrage, navigation, exploration guidée, demande à un expert) ou passive (voir point I.7.3.).

ARBAS96 offre un système de requêtes pour récupérer les informations liées à des problèmes de négociation. Les requêtes peuvent porter sur des problèmes, des acteurs, des propositions, des arguments, des buts, ..... Dans ARBAS96, il faut distinguer deux types de requêtes, celles qui sont effectuées par un opérateur humain et celles effectuées par le système lui-même. Nous détaillerons, plus loin, les requêtes proposées par ARBAS96 (voir point II.5.6.).

ARBAS96 offre aussi un système de filtrage qui permet de délimiter la recherche de problèmes particuliers. A chaque problème est associé un ensemble de mots-clés, un titre et une description générale de ce problème. Lorsqu'un utilisateur désire connaître tous les problèmes qui traitent d'un sujet particulier, il peut entrer des clés de recherche qui analyseront les mots-clés, le titre et l'énoncé des problèmes.



Dans ARBAS96, les utilisateurs peuvent récupérer des objets qui servent de références à une proposition, un argument ou un but. Ces objets peuvent appartenir à d'autres bases de données de l'organisation. ARBAS96 permet de naviguer entre des problèmes de négociation et des objets extérieurs sur lesquels ils travaillent.

Enfin, ARBAS96 peut offrir, grâce à son système d'enchaînement des états, une méthode d'exploration à travers des problèmes de négociation.

En conclusion, ARBAS96 offre différents modes de récupération des informations. Ces modes sont tirés de la classification établie par Morrison et Weiser (1996) (voir point I.7.3.).

### **La maintenance**

Au niveau de la maintenance du système, ARBAS96 offre différentes fonctions. D'abord, rappelons qu'ARBAS96 offre différents niveaux d'accès (public, autorisé et privé). De ce fait, une des fonctions de maintenance est liée à la gestion des accès.

ARBAS96 permet aussi de supprimer ou de modifier des propositions et des arguments. Ces fonctions peuvent être nécessaires pour le respect de la vie privée des individus ou pour la suppression d'informations obsolètes, par exemple.

De manière à gérer la mémoire organisationnelle liée à la négociation, ARBAS96 doit permettre de supprimer les éléments de la mémoire qui sont obsolètes. ARBAS96 peut détecter les anciens problèmes qui n'ont plus été consultés depuis un certain temps et proposer leur suppression.

## **II.5.5. Le fonctionnement général d'ARBAS96**

ARBAS a été étendu en termes de données et de fonctions de sorte que son fonctionnement s'en trouve modifié. Nous présentons le déroulement d'une négociation selon ARBAS96 à l'aide d'un graphe d'enchaînement des traitements (Figure II.9., II.10., II.11.) décomposé selon les trois phases de la négociation que nous avons présentés précédemment : la phase de pré-négociation, la phase de négociation proprement dite et la phase de post-négociation. Cette description résume tous les concepts que nous avons exposés précédemment et indique de quelle manière ils interviennent au cours d'une négociation.

## La phase de pré-négociation

La phase de pré-négociation a pour objectif d'établir une définition complète du problème de négociation à résoudre. La figure II.9. dissèque la manière dont une définition de problème est énoncée dans ARBAS96.

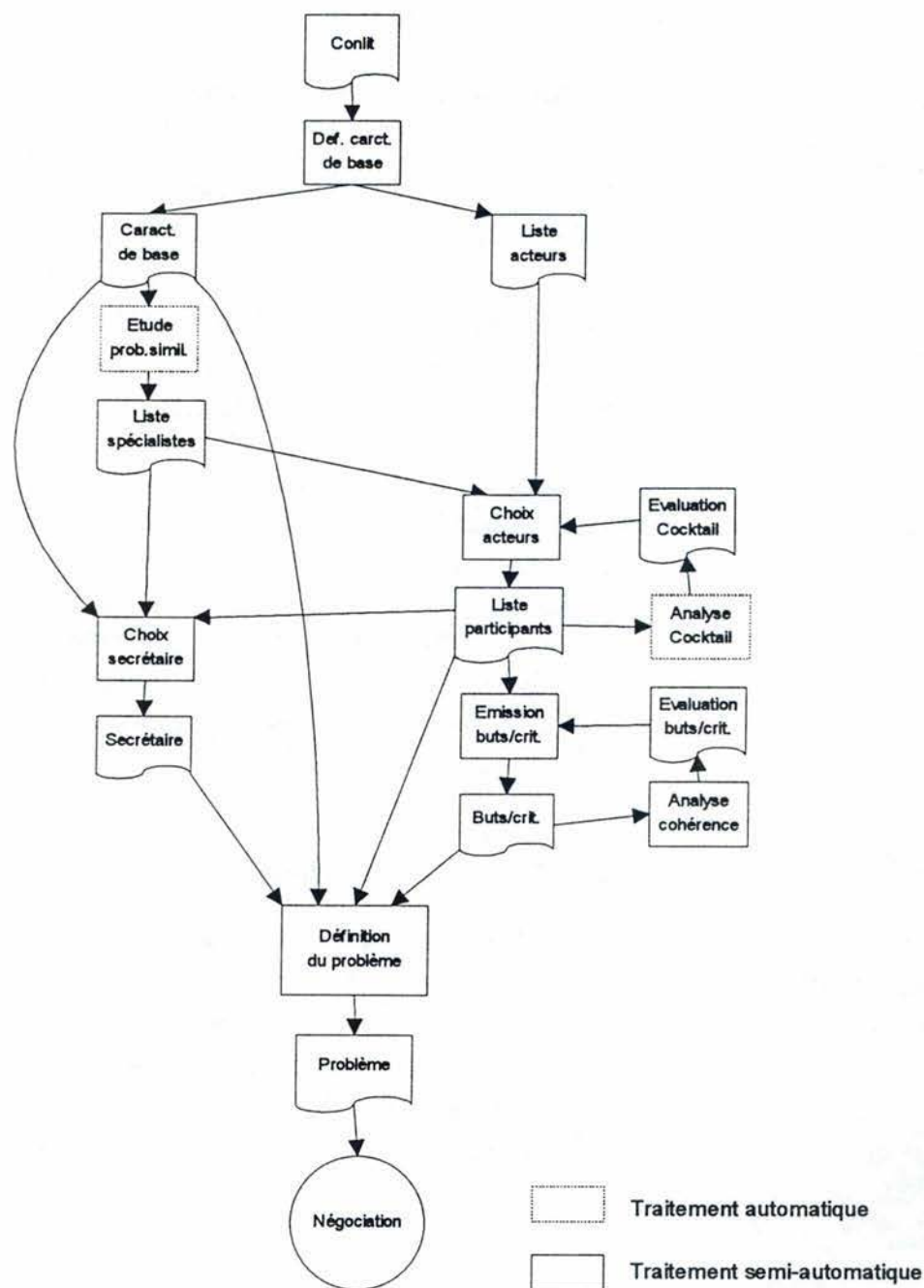


Figure II.9. : Phase de pré-négociation



A la naissance d'un conflit, les premières caractéristiques du problème peuvent être mémorisées: le titre du problème, la date du problème, l'énoncé du problème, les mots-clés du problème, une première liste d'acteurs et les événements déclencheurs éventuels. Les mots-clés sont particulièrement importants dans la mesure où ils permettent la recherche de cas similaires.

### **Le choix des acteurs**

Une première liste d'acteurs est énumérée. ARBAS96 fouille sa mémoire relative aux négociations et propose une liste de spécialistes. Cette liste est dressée sur base des problèmes antérieurs et comportant un certain nombre de similarités avec la situation courante. La position d'un acteur dans cette liste dépend du nombre de négociations auxquelles il a participé dans le secteur déterminé par les mots-clés. Le degré de spécialité d'un acteur est inféré à partir du degré de similitude des problèmes trouvés (i.e. pourcentage de mots-clés retrouvés dans le titre, la description et/ou les mots-clés). Il s'agit du type de recherche appliqué dans les bibliothèques où pour rechercher un livre sur base de mots-clés, le système analyse tous les livres qui retrouvent ces mots-clés, ou une partie de ceux-ci, dans leur titre, résumé et/ou mots-clés.

Cette liste va permettre d'aider à désigner un secrétaire et à compléter la liste des participants. En fait, ARBAS établit une liste de problèmes similaires et propose leur secrétaire.

ARBAS propose mais n'impose pas le secrétaire. Nous tenons à rappeler ici qu'ARBAS est un outil d'aide à la négociation et qu'il ne s'agit en aucun cas de considérer que le système puisse décider à la place de l'utilisateur. ARBAS est à la négociation ce que l'archet est au violoncelle. Il est possible de jouer sans, mais les résultats ne sont peut-être pas aussi probants.

ARBAS étudie les aptitudes des acteurs en se basant sur les négociations passées. Certains acteurs n'ont jamais émis de proposition gagnante, ne participent pas activement aux négociations, ou ont accepté des propositions qui se sont finalement avérées mauvaises (feed-back). ARBAS tente aussi d'éviter les associations infructueuses qu'il peut dégager des négociations passées. Certaines personnes ne sont jamais arrivées à s'accorder et il n'est pas recommandable de continuer à les opposer sous peine de transformer les négociations en conflits de personnalités. Pour faire connaître les risques de problème, ARBAS analyse le cocktail des participants et intervient en produisant des messages d'avertissement.

Dès que les acteurs de la négociation ont été désignés, ils doivent émettre les buts qu'ils espèrent voir se réaliser en cours de négociation. Ces buts sont évalués à l'aide de critères afin d'en relever les contradictions.

Une fois que l'ensemble des participants a émis ses buts et critères, et que le secrétaire a été désigné, il peuvent procéder à l'affinement de la définition du problème. Cette définition reprend les caractéristiques de base établies plus tôt et éventuellement modifiées par les participants. La définition de problème permet également aux acteurs d'établir le protocole de négociation. Il est choisi par le secrétaire, en accord avec les participants, et peut d'ailleurs être modifié en cours de négociation.

Les caractéristiques principales du problème de négociation sont enregistrées par ARBAS96 et la négociation peut donc commencer.

### **La négociation proprement dite**

Une fois le décor planté, la négociation peut commencer (Figure II.10.). Les acteurs autorisés peuvent émettre des propositions sur lesquelles les autres parties vont réagir en émettant des critiques (sous formes d'arguments et de contre-arguments) ou des nouvelles propositions.



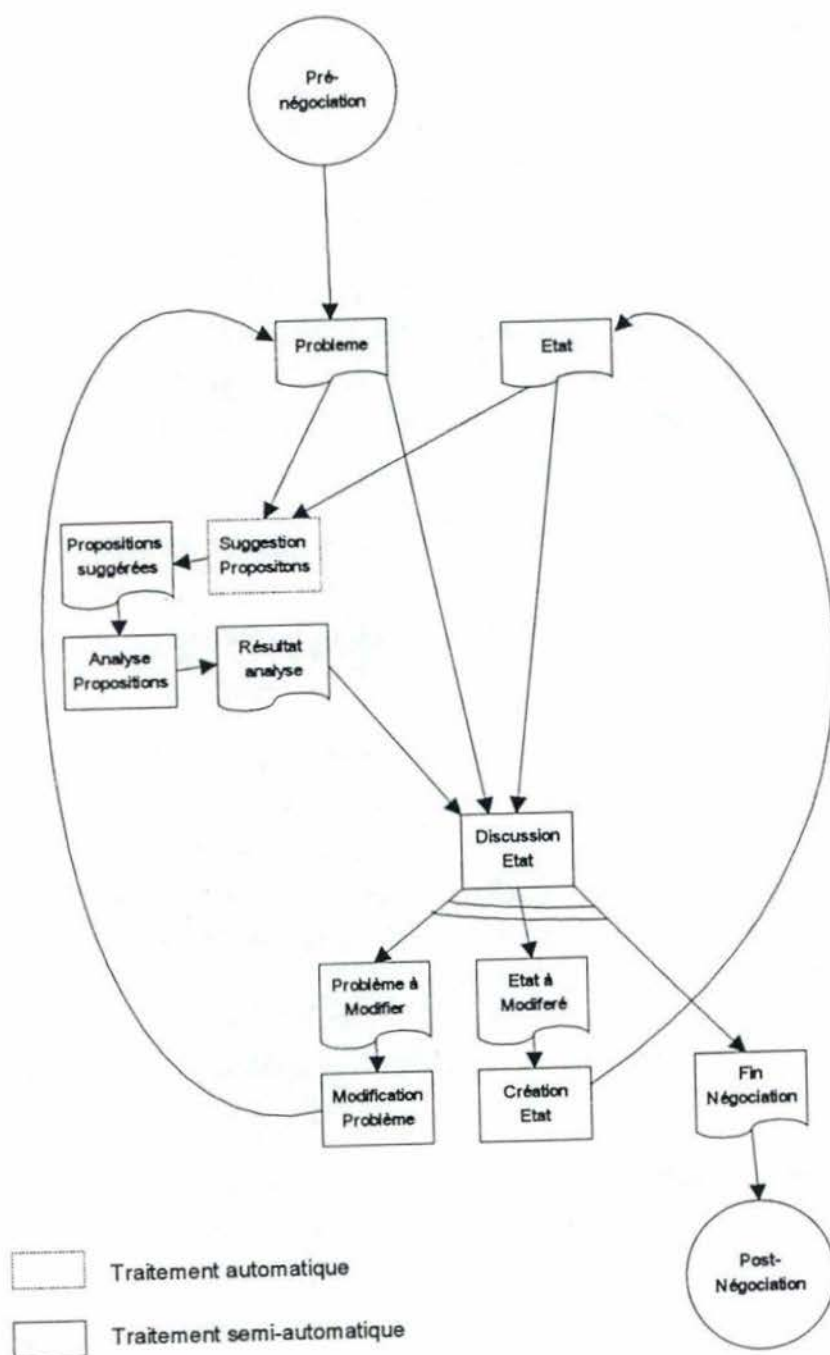


Figure II.10. : Phase de négociation

## **Les changements d'états**

ARBAS96 représente l'évolution du processus de négociation par une succession d'états. L'apparition d'un nouvel état est justifiée par l'émission et/ou la suppression d'une ou plusieurs propositions. Cette création d'un nouvel état se produit selon différents cas.

Le cas le plus courant de création d'un état est l'émission d'une nouvelle proposition par un acteur. L'état nouveau est constitué alors des anciennes propositions accompagnées de la nouvelle. Le cas où un acteur change une proposition est aussi un événement qui produit un nouvel état. Celui-ci reprend toutes les propositions de l'état précédent moins la proposition qui a été modifiée et plus la proposition qui la remplace. La suppression d'une proposition génère également un nouvel état. Celui-ci est composé des propositions du dernier état moins la proposition supprimée.

Une autre façon d'obtenir un nouvel état est de réaliser un tour de table. On demande aux acteurs qui le désirent d'émettre une proposition. Le nouvel état est alors composé de ces propositions, toutes les anciennes étant effacées. Le tour de table constitue un bon moyen pour rafraîchir une négociation.

## **Déroulement d'une négociation**

Après que le problème ait été défini, ARBAS, fouillant sa mémoire de cas, émet une liste triée de propositions émises dans le cadre de problèmes traitant de sujets semblables. Le tri des propositions s'effectue selon cinq critères :

- La similitude (mots-clés) entre le problème courant et ceux qui ont généré les propositions à trier
- Acceptation de la proposition comme solution (accord, imposé, temps dépassé, nombre de propositions dépassées). Ici interviennent aussi les arguments et contre-arguments pour mesurer le degré d'acceptation de la proposition.
- Feed-back sur l'éventuel accord.
- Similitude dans les buts des acteurs
- Les aptitudes des acteurs



Les propositions sont d'abord triées en fonction de la similitude entre le problème courant et ceux où elles ont été générées. Ensuite, les propositions sont triées en fonction du résultat du problème (réussies ou imposées), puis de l'évaluation qui est portée sur l'éventuel accord.

Les propositions de solutions sont présentées sous forme de liste, mais la sélection de l'une d'entre-elles déclenche l'ouverture du problème dans lequel elle a été générée. Le problème s'ouvre et son état courant correspond à l'état qui a été créé par l'émission de la proposition. Ainsi, l'utilisateur peut comprendre le contexte dans lequel la proposition a été développée. Dans la partie relative à la mémoire organisationnelle, nous avons insisté sur l'importance du contexte.

Donc, le secrétaire peut consulter les propositions qui ont été présentées dans ces différents problèmes et en choisir une pour première proposition de solution. Cette proposition, éventuellement corrigée par le secrétaire, est soumise au jugement des participants.

Remarquons que la première proposition peut aussi être émise par un autre acteur que le secrétaire de la négociation. La négociation proprement dite peut aussi commencer par un tour de table mais pour des raisons de facilités de cet exposé nous considérons que la résolution du problème commence par l'émission d'une seule proposition.

Chaque acteur va pouvoir critiquer la première proposition, c'est-à-dire donner son opinion vis-à-vis de l'action proposée. Il dit s'il la supporte ou s'y oppose et pourquoi. Les acteurs vont aussi pouvoir attaquer les différents arguments avancés par l'auteur de la proposition à l'aide de contre-arguments.

A partir de là, un (ou plusieurs) acteur(s) va (vont) pouvoir émettre une autre proposition donnant lieu à un état à deux (ou plus) propositions, chacune argumentée, contre-argumentée (éventuellement contre-contre-argumentée, ...). Lorsqu'une personne voit que sa proposition n'est plus viable car elle n'est pas soutenue ou parce que ses arguments sont contrés, elle peut décider d'abandonner sa proposition ou d'en émettre une nouvelle. Toutes les propositions restantes forment le nouvel état de la négociation. Toute personne peut ainsi visualiser chaque proposition, argument, et formuler sa proposition de manière à rencontrer les attentes d'un maximum d'acteurs ce qui permettra de se rapprocher d'une solution acceptable.

Il est aussi possible d'envisager le cas où un acteur désire émettre deux propositions différentes. Il a deux actions à proposer pour résoudre le problème, chacune étant défen-

dues par une série distincte d'arguments. Au cours de la négociation, il est probable qu'il en abandonnera une au profit de l'autre selon l'évolution de la discussion.

Lorsqu'un acteur désire émettre une proposition, il peut se servir de la mémoire organisationnelle pour voir quelles sont les propositions qui ont été fructueuses dans le passé et les remanier dans le cadre de la négociation actuelle. A ce propos, nous rappelons qu'ARBAS96 propose une liste de propositions émises dans le cadre de problèmes similaires. Cependant, un utilisateur est libre d'effectuer toute requête qu'il juge intéressante pour améliorer la qualité de sa proposition. Par exemple, si un acteur désire faire avancer la négociation en allant dans la direction des autres participants, il peut rechercher des propositions émises dans le passé par des acteurs ayant des buts similaires à ceux des participants actuels. Nous remarquons aussi qu'un acteur peut, à l'aide d'objets de référence, mettre plus de poids dans une proposition ou un argument. Ces objets de référence créent des liens entre la négociation et des sources d'information externes.

En cours de négociation, le contexte peut évoluer avec l'apparition (ou la disparition) d'un ou plusieurs acteurs. Lors de l'arrivée d'un nouvel acteur, ARBAS analyse, comme au moment de la définition du problème, les caractéristiques du nouveau venu et soumet son impression au secrétaire si cela s'avère nécessaire. Les acteurs peuvent aussi modifier leurs buts et les critères de ces buts au cours de la négociation. Un autre exemple de facteur qui modifie le contexte de la négociation est le protocole.

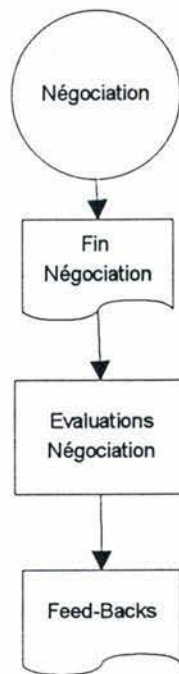
ARBAS96 gère aussi les situations de blocage qui apparaissent dans les négociations. En fait, lorsque ARBAS96 constate qu'aucun mouvement (nouvelle proposition ou nouvel argument) n'a été opéré depuis un certain temps (via un contrôle des moyennes), il émet un message qui propose au secrétaire d'effectuer un tour de table ou de changer les acteurs de la négociation si cela est possible.

A la clôture de la négociation, ses conditions de terminaison sont enregistrées : résolution du conflit (proposition acceptée par tous les acteurs), temps imparti écoulé (abandon de la négociation), nombre maximum de propositions émises dépassé (abandon de la négociation), majorité qui supporte une proposition (solution imposée). Lorsque les acteurs sont arrivés à une solution, un accord peut être ratifié.

## **La Post-négociation**

La phase de post-négociation (Figure II.11.) gère les fonctions qui peuvent être déclenchées une fois que la négociation est terminée.





*Figure II.11. : Phase de post-négociation*

Au cours de la négociation les acteurs ont émis leurs avis en proposant des solutions de solutions ou en posant des arguments. Ces informations sont utiles car elles constituent un moyen pour évaluer la solidité des propositions et des arguments. Une fois que la négociation est terminée, il est intéressant de demander aux acteurs de poser une analyse critique sur l'accord qui a été établi. Cette analyse permet de faire comprendre, par exemple, quelles sont les difficultés auxquelles il a fallu faire face pour mettre en place la proposition. Cette analyse est un élément supplémentaire qui permet d'évaluer une proposition et qui peut, dès lors, servir pour l'apprentissage et la réutilisation des négociations antérieures.

## **Méthode d'évaluation d'ARBAS96**

ARBAS96 possède différents éléments qui lui permettent d'intervenir de manière intelligente, active, au sein d'une négociation.

Pour agir de manière active, ARBAS96 dispose d'informations sur les négociations telles que les propositions, les buts, les enchaînements de propositions, les arguments,

ainsi que des informations sur les acteurs. Ce sont ces différentes informations qui permettent à ARBAS96 d'agir de manière active dans une négociation.

ARBAS96 peut évaluer les acteurs, les propositions et les arguments.

L'évaluation d'un acteur se base sur trois critères : son expérience, son degré de participation et la qualité de ses interventions.

- L'expérience d'un acteur est évaluée sur base du nombre de négociations auxquelles il a participé. ARBAS96 peut aussi déterminer l'expérience d'un acteur dans un domaine particulier limité par les mots-clés.
- Le degré de participation d'un acteur est évalué sur base du nombre moyen de propositions et/ou d'arguments émis dans une ou plusieurs négociations.
- Enfin, la qualité des interventions d'un acteur se réfère à la qualité de ses propositions et de ses arguments.

L'évaluation de la qualité d'une proposition se base sur l'intonation avec laquelle elle a été émise (e.g. conviction), du feed-back qui lui a éventuellement été porté lorsqu'elle a donné lieu à un accord, de l'évaluation de ses contre-propositions et de l'évaluation de ses arguments (positifs ou négatifs). Dans ARBAS96, nous évaluons les propositions par des poids. Le poids d'une proposition est la différence entre les poids de ses arguments de support et les poids de ses arguments d'opposition, augmentée du coefficient d'intonation (Sentiment=1, Opinion=2, Croyance=3, Conviction=4, Persuasion=5) et du poids des feed-backs (Very Bad=-2, Bad = -1, Good=1, Very Good=2, Perfect=3), et diminuée du poids de ses contre-propositions.

**Poids (proposition) =  $\Sigma$  Poids (Arg\_positif) -  $\Sigma$  Poids (Arg\_négatif) + coefficient d'intonation -  $\Sigma$  Poids (contre\_prop) +  $\Sigma$  Poids (Feed-back)**

où Arg\_positif est un argument de type *support*

et Arg\_négatif est un argument de type *oppose*

L'évaluation d'un argument se base sur l'intonation avec laquelle il a été émis par son auteur et sur l'évaluation des arguments (positifs ou négatifs) portés sur cet argument. Dans ARBAS96, nous évaluons les arguments par des poids. Le poids d'un argument est la différence entre les poids des arguments de support et les poids des arguments d'op-



position, augmentée du coefficient d'intonation (Sentiment=1, Opinion=2, Croyance=3, Conviction=4, Persuasion=5).

$$\text{Poids (Argument)} = \Sigma \text{Poids (Arg\_positif)} - \Sigma \text{Poids (Arg\_négatif)} + \text{coefficient d'intonation}$$

où Arg\_positif est un argument de type *support*

et Arg\_négatif est un argument de type *oppose*

Nous tenons à préciser que dans les mesures de poids des arguments ou des propositions, nous avons pris la liberté de donner des coefficients d'évaluation aux feed-backs et aux intonations. Ces coefficients pourraient être modifiés si l'on considère, par exemple, que le feed-back d'une proposition est beaucoup plus important que ses arguments. De plus, nous reconnaissons la nature intuitive de l'élaboration de ces formules. Ces dernières ont surtout pour objectif de relever les critères d'évaluation des éléments d'une négociation. Une recherche plus approfondie devrait pouvoir être menée.

### II.5.6. Mise en oeuvre d'ARBAS96

ARBAS96 est constitué de quatre composants principaux (Figure II.12.):

- Une plate-forme de communication et de collaboration qui permet aux individus d'accéder et d'argumenter dans une négociation. Cette infrastructure répond aux exigences d'intégration spatiales et temporelles. Chaque participant doit pouvoir se connecter depuis son lieu de travail et argumenter comme s'il se trouvait assis à la table de négociation. L'intégration temporelle implique, pour sa part, que l'acteur puisse accéder à la négociation quand bon lui semble et qu'il puisse consulter des informations du passé.
- Une base de données définie pour supporter le schéma Entité-Association d'ARBAS96 (voir point II.5.2.).
- Un système de requêtes permettant d'accéder aux informations de la base de données.
- Une interface Homme/Machine spécialement étudiée pour être utilisée par les acteurs des négociations qui ne seront pas nécessairement familiers avec ce genre d'outils.

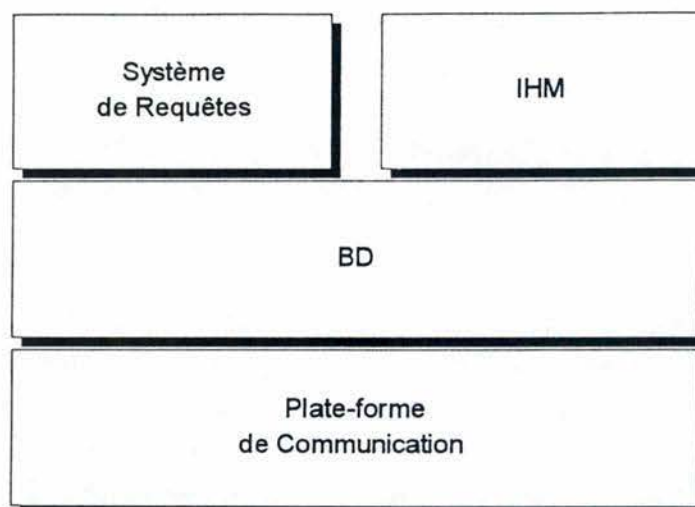


Figure II.12. : Composants principaux d'ARBAS96

Nous allons étudier plus en profondeur les composants *Systèmes de Requêtes* et *IHM* de cette architecture logique. Les deux autres composants n'abordant pas des aspects qui nous intéressent dans le cadre de ce mémoire.

### Le système de requêtes

Les données acquises au cours de la négociation sont stockées dans la mémoire organisationnelle. ARBAS permet de récupérer les informations qui sont enfouies dans la mémoire. Nous exprimons les récupérations d'information à l'aide de pseudo-requêtes SQL.

Ces pseudo-requêtes sont composées du mot réservé *select* qui détermine le domaine sur lequel la recherche sera portée et du mot réservé *where* qui détermine le critère de la recherche (optionnel) sous la forme d'une expression booléenne. Cette expression booléenne est elle-même composée d'expressions booléennes plus petites portant sur des sujets particuliers (problème, acteur, mots-clés, ....). Nous reprenons ci-après des exemples de ce genre d'expressions :

<Expr\_problème> est une expression booléenne portant sur les problèmes.

Ex : problème='USA/CANADA' or problème='Software'



<Expr\_mots\_clés> est une expression booléenne portant sur les mots-clés des problèmes.

Ex : mot\_clé='Lumber' or mot\_clé='bois'.

<Expr\_acteurs> est une expression booléenne portant sur les acteurs.

Ex : acteur='USA' and acteur='CANADA'.

<Expr\_date> est une expression booléenne portant sur les dates.

Ex : date < '28/12/95'.

<Expr\_objectifs> est une expression booléenne portant sur les objectifs.

Ex : objectif='bénéfice' or objectif='honnêteté'

<Expr\_buts> est une expression booléenne portant sur les buts.

Ex : but='marché équitable'

<Expr\_critère> est une expression booléenne portant sur les critères de satisfaction des buts.

Ex : critère='équité'

<Expr\_intonation> est une expression booléenne portant sur l'intonation d'une proposition ou d'un argument.

Ex : intonation='Persuasion' or intonation='Conviction'

<Expr\_type> est une expression booléenne portant sur le type d'un argument

Ex : type='Oppose' or type='Don't care'

A présent, nous allons nous intéresser à détailler les caractéristiques des différentes requêtes présentes dans ARBAS96. Nous nous posons les questions suivantes :

- **Qui** est susceptible d'utiliser les requêtes ?
- **Quand** une requête est-elle lancée ?
- **Pourquoi** est-elle exécutée ?
- **Comment** est-elle mise en route ?

*Qui est susceptible d'utiliser les requêtes ?*

Dans ARBAS96, les requêtes sont utilisées, aussi bien par l'utilisateur du système, que par le système lui-même. En effet, ARBAS96 a été conçu de manière à fournir un support intelligent aux utilisateurs du système. Dans certaines conditions, ARBAS96 intervient de sa propre initiative pour améliorer le déroulement de la négociation. Sycara (1993) définit les conditions à satisfaire pour qu'un système d'aide à la négociation soit intelligent :

- Le système doit proposer au fur et à mesure des modifications à une proposition pour aider les acteurs à restreindre leurs divergences de vues.
- Les acteurs doivent réagir de manière positive ou négative aux propositions de solution. Ces réactions doivent être mémorisées dans la mémoire organisationnelle pour servir de moyen d'évaluation de la proposition.
- Le système doit pouvoir évaluer dans quelle mesure toute nouvelle proposition fait avancer le conflit vers sa solution.
- Les acteurs doivent pouvoir communiquer leurs arguments et justifications de leur position dans le problème.

ARBAS96 ne répond pas entièrement aux exigences d'un système intelligent, mais possède tout de même des caractéristiques d'un tel système. Le défaut actuel d'ARBAS96 réside dans son manque d'analyse des problèmes, propositions et arguments. Une solution serait d'utiliser la technique du case-based reasoning. Sycara (1993) expose différentes idées par rapport à cette technique. Le case-based reasoning est un paradigme intelligent de résolution de problèmes qui permet à un système d'exploiter ses expériences et d'incorporer automatiquement des nouvelles expériences dans sa mémoire de cas. Le case-based reasoning utilise la connaissance de prises de décision passées pour proposer des solutions à un nouveau cas concernant le même sujet. La connaissance de cas à succès peut être utilisée pour des négociations ultérieures, tandis que celle des cas d'échecs peut servir pour avertir les acteurs d'un éventuel risque de déconvenue (Sycara (1993)). Liang (1993) considère le raisonnement par analogie comme un mécanisme d'inférence qui permet d'identifier des similarités entre les cas et de développer de nouvelles solutions. La carence d'ARBAS96 réside essentiellement dans son absence de modèle d'interprétation intelligente des cas présents dans la mémoire. Il mémorise des cas, mais il n'est pas (encore) muni de mécanismes qui lui permettent de comparer deux problèmes, propositions ou arguments différents de manière à pouvoir en tirer les similitudes.



Outre le système, les utilisateurs peuvent consulter la mémoire organisationnelle liée à la négociation. Les utilisateurs peuvent être des acteurs d'une négociation actuelle qui, d'une part, désirent mieux comprendre la négociation à laquelle ils participent et, d'autre part, recherchent des bonnes idées en consultant les négociations passées. Les utilisateurs peuvent aussi être des personnes recherchant des informations sur des données plus générales (statistiques) liées aux négociations (e.g. quelles sont les négociations auxquelles tel acteur a participé, quels sont les problèmes de négociation auxquels l'organisation a été confrontée, etc...).

### *Quand une requête est-elle lancée ?*

Lorsqu'elles sont destinées à répondre à la demande d'un utilisateur avide d'informations sur les négociations passées ou présentes, les requêtes sont lancées au moment où l'utilisateur le désire.

Des requêtes peuvent aussi être mises en oeuvre pendant le déroulement d'une négociation. Les différents acteurs de la négociation peuvent à tout moment interroger la mémoire organisationnelle pour obtenir les renseignements dont ils ont besoin pour mieux comprendre leur comportement et celui des autres acteurs, pour améliorer la qualité de leurs actions, ou pour s'assurer de la compatibilité de leurs buts.

Dans ARBAS96, les trois moments fondamentaux sont la définition du problème, l'émission d'une proposition et l'émission d'un argument. A chacune de ces étapes, la mémoire a de grandes chances d'être consultée.

Au moment de la définition d'un problème, le système lance des requêtes qui scannent sa mémoire à la recherche de cas similaires et dont il présente la solution quand elle existe. Le système exécute aussi des requêtes lorsqu'il désire évaluer les performances des acteurs.

### *Pourquoi exécuter les requêtes ?*

Chacune des requêtes que nous détaillerons ci-après répond d'une manière ou d'une autre aux exigences de la mémoire et de l'apprentissage organisationnels. Nous relevons aussi les requêtes qui permettent d'augmenter l'intelligence d'ARBAS96.

## **Recherche des problèmes :**

Select Problème

```

[Where
    [<Expr_mots_clés>]
[and/or]
    [<Expr_raison_fin]
[and/or]
    [<Expr_acteurs>]
[and/or]
    [<Expr_date>]
[and/or]
    [<Expr_objectifs>]
]

```

Cette requête peut servir de support à la définition d'un nouveau problème. Elle permet aussi la recherche d'informations qui pourrait aider les acteurs à émettre des propositions ou arguments pertinents.

La recherche de problèmes peut se faire selon différents critères, à savoir :

- **sur mots-clés** : retourne les problèmes concernant un ou plusieurs sujets.

Cette requête peut répondre aux besoins du système intégratif temporel du modèle de Stein et Zwass. Cette requête fouille la mémoire de l'organisation pour voir si un problème relatif à certains sujets n'a pas encore été résolu par l'organisation. Une fois ce(s) problème(s) trouvé(s), il est possible de s'en servir pour mieux résoudre les problèmes actuels en tirant des leçons du passé. En utilisant des mots-clés relatifs à des conflits de personnel, il est aussi possible d'inférer le climat général dans lequel baigne l'organisation (système de Maintenance des procédures de travail). Par exemple, si le mot-clé est 'grève' et qu'il y a beaucoup de problèmes répondant à ce mot-clé, on pourra en déduire que le climat n'est pas spécialement bon.



La requête sur mots-clés peut aussi supporter les besoins du système Adaptatif. En effet, en recherchant tous les problèmes relatifs à un sujet, il est possible de voir de quelle manière ils ont été résolus par l'organisation, c'est-à-dire de quelle manière l'organisation s'y est adaptée.

- **sur raison-fin** : retourne les problèmes qui répondent à certaines conditions de terminaison (réussite, solution imposée, abandon faute de temps, abandon car trop de propositions émises).

Cette requête est utile car elle permet de retrouver tous les échecs ou réussites d'une organisation en matière de négociations. Ces informations peuvent servir d'apprentissage pour l'organisation.

- **sur acteur** : retourne les problèmes dans lesquels ont participé un ou plusieurs acteurs.

Cette requête peut servir de support au sous-système de Maintenance des procédures de travail puisqu'elle permet de récupérer une partie de l'historique de travail des individus. Elle permet d'obtenir une description des projets auxquels des individus ont travaillé. En connaissant les domaines de prédilection d'un acteur, le système peut automatiquement proposer la candidature de cet acteur pour la résolution d'une négociation. De même, il peut aussi déconseiller la participation d'un acteur au sein d'une négociation car celui-ci ne s'est pas montré efficace lors de précédentes résolutions de conflits. En posant la requête sur les acteurs et les conditions de terminaison d'une négociation, le système peut évaluer les performances des acteurs. ARBAS96 possède des informations qui sont utiles pour la composition des participants d'une négociation.

- **sur date** : retourne les problèmes qui ont été négociés à une date précise ou entre deux dates.

Cette requête répond aux besoins du sous-système Intégratif temporel.

- **sur objectif** : retourne les problèmes concernant un ou plusieurs objectifs organisationnels.

Cette requête sert de support au sous-système de réalisation de but pour évaluer le progrès en direction de la réalisation de l'objectif. Si à un objectif particulier de l'organisation correspond beaucoup de problèmes, on pourra en déduire que la réalisation de cet objectif n'est pas aisée et que l'on est peut être encore loin de sa réalisation. Il faudra, dès lors, peut-être revoir cet objectif en le modifiant ou tout simplement en le supprimant. De ce fait, ARBAS96 peut servir d'outil à l'évaluation de la réalisation des objectifs stratégiques de l'organisation.

Les recherches sur problèmes retournent des listes de problèmes. En sélectionnant l'un d'entre eux, toutes ses caractéristiques sont présentées. Le protocole de négociation est une caractéristique importante car elle reflète le contexte selon lequel les acteurs ont pu intervenir dans la négociation.

#### Recherche des acteurs :

```
Select Acteur
[Where
    [<Expr_problème>]
[and/or]
    [<Expr_mots_clés>]
[and/or]
    [<Expr_date>]
]
```

La recherche des acteurs peut se faire selon différents critères, à savoir :

- **sur problème :** retourne les acteurs qui ont participé à un ou plusieurs problèmes.



Cette requête reflète l'intégration spatiale d'un problème. Elle retourne tous les acteurs impliqués à un moment donné dans le problème ce qui définit l'espace concerné par le problème.

- **sur mot-clé :** retourne les acteurs qui ont participé à des problèmes concernant un ou plusieurs sujets.

Cette requête permet de voir quels sont les acteurs habitués à tel type de problème. Il est ainsi possible de connaître quelles sont les spécialités des acteurs (sous-système de Maintenance des procédures de travail ) et le système pourra se servir de cette information pour proposer la participation d'un acteur dans une négociation.

- **sur date :** retourne les acteurs qui ont participé à des problèmes entre deux dates.

Cette requête répond aux besoins du sous-système Intégratif temporel.

Les requêtes sur acteurs retournent des listes d'acteurs. En sélectionnant l'un d'entre eux, toutes ses caractéristiques sont présentées. Il est, par exemple, intéressant de savoir quelle fonction occupe un acteur au moment où il intervient dans une négociation, car son comportement est, en partie, justifié par la position qu'il occupe dans l'organisation.

#### Recherche des buts :

```
Select But
[Where
    [<Expr_problème>]
[and/or]
    [<Expr_acteur>]
]
```

La recherche des buts peut se faire selon différents critères, à savoir :

- **sur problème** : retourne les buts qui ont été présentés au cours d'un ou plusieurs problèmes.
- **sur acteur** : retourne les buts d'un ou plusieurs acteurs.

Cette requête permet de mieux comprendre le comportement général présenté par un ou plusieurs individus au cours d'une ou plusieurs négociations. Elle répond aux exigences du système de Maintenance des procédures de travail du modèle de Stein et Zwass.

#### Recherche des critères :

```
Select Critère
[Where
    <Expr_problème>
[and/or]
    [<Expr_acteur>]
[and/or]
    [<Expr_but>]
]
```

La recherche des critères peut se faire selon différents angles, à savoir :

- **sur problème** : retourne les critères qui ont été présentés au cours d'un ou plusieurs problèmes.



Cette requête reflète les différents facteurs auxquels les acteurs ont attaché de l'importance au cours de différents problèmes. Lorsque cette requête porte sur plusieurs problèmes, elle peut aussi bien retourner l'union des critères présents dans ces problèmes, que l'intersection de ces critères. Si deux problèmes se partagent un critère, il est possible de se servir de l'expérience acquise pour le premier au profit du deuxième. Par exemple, si l'honnêteté constitue un critère d'un acteur, logiquement, ses arguments doivent être liés à cette ligne de conduite et ils peuvent être utilisés par un autre acteur lors d'une autre négociation (apprentissage).

- **sur acteur :** retourne les critères d'un ou plusieurs acteurs.

Cette requête permet de mieux comprendre le comportement général présenté par un ou plusieurs individus au cours d'une ou plusieurs négociations. Elle répond aux exigences du système de Maintenance des procédures de travail du modèle de Stein et Zwass. L'analyse de l'importance du critère pour un individu permet aussi d'évaluer les chances de réussite d'une négociation. Par exemple, si des acteurs défendent corps et âmes des critères opposés, il y a peu de chance que le problème de négociation soit résolu. De ce fait, cette requête répond aussi aux exigences du système de réalisation des buts du modèle de Stein et Zwass.

- **sur buts :** retourne les critères d'un ou plusieurs buts.

Cette requête permet d'analyser les critères qui sont mis en relation avec des buts. Il est ainsi possible, lorsqu'un acteur émet un but, de lui proposer différents critères de succès établis dans le passé pour des buts similaires. Il faut aussi déconseiller aux acteurs de présenter des critères qui se sont avérés inefficaces dans le passé.

#### Recherche des états :

Select Etat

[Where

<Expr\_problème>

[and/or]

```

[<Expr_date>]
[and/or]
[<Expr_prop>]
]

```

La recherche des états peut se faire selon différents critères, a savoir :

- **sur date** : retourne les états d'un problème entre deux dates ou à une date précise.

Cette requête répond aux besoins du sous-système Intégratif temporel. Elle permet de "voyager" dans le temps et de comprendre l'évolution d'une négociation dans une fourchette de temps.

- **sur proposition** : retourne les états d'un problème entre deux propositions ou l'état qui a été créé par l'apparition d'une nouvelle proposition.

Cette requête permet aussi de voyager dans le temps. En fait, elle permet de positionner l'état courant de la négociation au moment où une proposition a été diffusée. En voyageant dans le temps, les utilisateurs d'ARBAS96 peuvent comprendre exactement dans quel contexte une proposition ou un argument ont été émis. C'est le système d'états qui permet de reproduire ce contexte.

#### Recherche des propositions :

```

Select Proposition
[Where
    <Expr_problème>
[and/or]
    [<Expr_date>]
[and/or]

```



[<Expr\_acteur>]  
 [and/or]  
 [<Expr\_intonation>]  
 [and/or]  
 [<Expr\_type>]  
 [and/or]  
 [<Expr\_prop>]  
 ]

La recherche des propositions peut se faire selon différents critères, à savoir :

- **sur date** : retourne les propositions émises entre deux dates ou à une date précise.

Cette requête répond aux besoins du sous-système Intégratif temporel.

- **sur acteur** : retourne les propositions émises par un ou plusieurs acteurs.

Cette requête permet de mieux connaître le(s) acteur(s) et répond dès lors aux besoins du sous-système de Maintenance des procédures de travail . En analysant les propositions d'un acteur, il est possible de voir s'il est motivé (bonne participation), s'il est qualifié (bonne qualité des propositions), s'il est efficace (proposition=solution). En analysant les intonations de ses propositions, il est aussi possible de comprendre le caractère de l'individu : quelqu'un dont les propositions sont toujours accompagnées d'une intonation forte (e.g. Conviction) pourra être considéré comme une personne décidée. Par contre, lorsqu'un acteur ne pose que des propositions à faible degré d'intonation, il pourra être considéré comme peu sûr de lui. En résumé, cette requête offre un moyen aux utilisateurs du système de saisir le comportement d'un individu. Dans le courant d'une négociation, les participants peuvent facilement percevoir à qui ils ont affaire.

- **sur intonation** : retourne les propositions ayant une intonation donnée.

Cette requête peut être utile pour percevoir le climat qui règne entre les acteurs liés au problème (sous-système de Maintenance des procédures de travail). Si plusieurs propositions ont pour intonation "persuasion", cela voudra dire que les acteurs sont motivés à défendre leur proposition et qu'il sera difficile de les faire changer d'avis, la tension entre les acteurs peut alors être considérée comme forte. Dans de telles situations, le système avertit le secrétaire de la négociation et préconise un changement d'acteurs ou un tour de table pour faire bouger les choses. On pourra aussi se servir de ces informations pour mesurer le degré de résolution du problème, et indirectement du degré de réalisation des buts (sous-système de Réalisation de buts).

- **sur proposition** : retourne les propositions qui ont trouvé une origine positive ou négative dans une autre proposition.

Cette requête permet de mieux comprendre l'enchaînement des propositions au cours d'une négociation. Elle permet au système et aux utilisateurs d'apprendre quelle réaction a été posée face à une proposition particulière. Cette requête permet, en partie, de comprendre le pourquoi de l'émission d'une nouvelle proposition.

- **sur type** : retourne les propositions ayant un type donné.

Cette requête affine la précédente en ne retenant que les influences positives ou négatives entre deux propositions. Cette requête est intéressante pour l'apprentissage car elle représente une amélioration de la proposition. Dans le cas où une proposition a été émise avec une influence positive par rapport à une autre proposition, cette dernière peut être comme une bonne proposition mais qui a nécessité quelques améliorations. Par contre, lorsqu'une proposition est émise de manière à contrer une autre proposition, cette dernière reçoit une mauvaise évaluation. Le principe de liens entre les différentes propositions est un moyen pour acquérir l'avis (le feed-back) des acteurs par rapport aux propositions.

Les requêtes sur propositions vont retourner une liste de propositions.

#### **Recherche des arguments :**

Select Argument



```

[Where
    <Expr_problème>
    [and/or]
        [<Expr_proposition>]
    [and/or]
        [<Expr_date>]
    [and/or]
        [<Expr_acteur>]
    [and/or]
        [<Expr_intonation>]
    [and/or]
        [<Expr_argument>]
    [and/or]
        [<Expr_type>]
]

```

La recherche des arguments peut se faire selon différents critères, à savoir :

- **sur proposition** : retourne les arguments avancés dans une proposition.

Cette requête offre un moyen aux utilisateurs du système ainsi qu'à celui-ci d'évaluer la qualité d'une ou plusieurs propositions. Cette évaluation permet de savoir s'il est intéressant ou pas de réutiliser cette proposition pour une autre négociation.

- **sur date** : retourne les arguments avancés entre deux dates ou à une date précise.

Cette requête répond aux besoins du sous-système Intégratif temporel.

- **sur acteur** : retourne les arguments avancés par un ou plusieurs acteurs.

Au même titre que pour la recherche des propositions, cette requête permet de percevoir le comportement général des acteurs.

- **sur intonation** : retourne les arguments ayant une intonation donnée.

Au même titre que la requête des propositions correspondantes, cette requête peut être utile pour percevoir le climat qui règne entre les acteurs liés au problème (sous-système de Maintenance des procédures de travail ). Ces informations peuvent aussi servir pour mesurer le degré d'avancement d'un problème, et indirectement le degré de réalisation des buts (sous-système de Réalisation de buts).

- **sur argument** : retourne les arguments qui renforcent, détruisent ou conservent la validité d'un argument.

Cette requête permet de mesurer la valeur d'un argument. Si un argument est victime de nombreuses attaques (contre-argument), il ne sera pas très solide. Par contre, lorsqu'un argument est supporté par de nombreux autres arguments, il aura de la valeur. Cette requête offre une vue globale sur un argument et indique quelles sont ses attaques et quels sont ses supports ou avis neutres.

- **sur type** : retourne les propositions ayant un type donné.

Cette requête affine la précédente en ne retenant que les influences positives, négatives ou neutres posées sur un argument. Comme pour les propositions, elle permet de mieux saisir la manière dont un argument est considéré par les autres acteurs d'une négociation.

Les requêtes sur arguments vont retourner une liste d'arguments.

En conclusion, nous avons repris les requêtes qui sont présentes dans ARBAS96. A l'aide de ces requêtes, il est possible d'obtenir un grand nombre d'informations qui peuvent supporter les négociations actuelles ou futures. Dans la section suivante II.3.5., nous décrivons l'implémentation d'ARBAS96 incluant une présentation de l'architecture physique.



## L'interface Homme/Machine

L'interface constitue un problème clé d'ARBAS. Meinadier (1991) considère la qualité de l'interface utilisateur comme un facteur déterminant pour la facilité d'apprentissage, la facilité et l'efficacité d'utilisation. Les qualités de l'interface se traduisent directement en termes d'acceptation, de satisfaction, de productivité pour l'utilisateur et de rentabilité pour l'entreprise.

Afin d'intervenir de façon efficace et de formuler des propositions intéressantes, les acteurs doivent avoir une vue complète sur les éléments des problèmes en cours. Il est donc nécessaire de pouvoir visualiser facilement tous les éléments des problèmes courants. Il faut aussi permettre à tout le monde de venir consulter les problèmes actuels et anciens. Toutes ces informations collectées vont permettre d'augmenter la mémoire organisationnelle de l'entreprise en matière de négociations.

Le point de départ de la conception de l'interface d'ARBAS96 est une critique de la première version d'ARBAS.

L'interface graphique proposée dans la première version d'ARBAS est séduisante dans la mesure où elle permet d'accéder rapidement aux vues posées sur les propositions d'actions (voir point II.3). L'enchaînement des propositions est représenté par une séquence d'icônes ce qui constitue un bon moyen pour afficher l'évolution du problème. Pour obtenir les renseignements sur une proposition, il suffit d'effectuer un clique à l'aide de la souris sur l'icône relative à la proposition et une fenêtre apparaît avec les informations de celle-ci. Nous proposons, toutefois, d'aller plus loin dans l'utilisation des icônes. En effet, nous estimons que la représentation des vues par une fenêtre ne synthétise pas suffisamment tous les renseignements liés à une proposition d'action. Par exemple, pour connaître qui supporte ou s'oppose à une proposition, il faut passer en revue toutes les vues sur la proposition. Dès lors, nous proposons de représenter les arguments d'une proposition par un graphique constitué d'icônes et de relations entre ces icônes. Dans la partie gauche d'une fenêtre sont situés, en colonne, des arguments qui supportent (rectangle), détruisent (ellipse) ou maintiennent (triangle) la validité de la **proposition** et sur lesquels, en ligne, sont attachés d'autres arguments positifs, négatifs ou neutres. Le lecteur trouvera une illustration de cette représentation dans l'étude de cas.

Tout ce qui est important est représenté sur le graphique et les détails de chaque argument s'obtiennent en cliquant sur leur icône. Cette modélisation des arguments possède des similitudes avec celle proposée par Chang et Han (1995) et offre un moyen d'évaluation des arguments qui surgissent dans une négociation.



Nous représentons les états par des cartouches dont l'en-tête est composée par la date de création de l'état. Les états s'enchaînent et si une proposition est émise, suite à une influence positive ou négative d'une autre proposition, un lien est établi entre ces deux propositions. Un signe + ou - résumant le type d'influence exercée par la première proposition sur la deuxième.

Dans ARBAS96, nous avons voulu faciliter l'accès aux informations les plus consultées au cours d'un problème. L'utilisation d'icônes offre une productivité élevée (Bodart et al. (1994)). Nous appelons les requêtes effectuées à l'aide d'icônes, les **requêtes directes**.

Pour l'ensemble des requêtes directes, nous avons aussi songé à utiliser le principe de la métaphore. Notre idée consistait à représenter la table de la négociation autour de laquelle différents acteurs discutaient. Chacune des propositions se trouvait rangée dans un dossier placé devant l'acteur qui les avaient émises. Nous avons même pensé créer une bibliothèque contenant les différents problèmes de négociation. Cette idée semble très attractive mais souffre néanmoins de limites non négligeables. D'abord, elle n'offre pas une vue d'ensemble telle que celle soutenue par l'enchaînement des états. Il n'est pas facile de savoir quel est l'acteur qui a émis la dernière proposition, quels sont les liens entre les différentes propositions, etc... . De plus, dans les négociations, il est reconnu qu'il faut séparer les personnalités du problème (Nunamaker et al. (1991)) et le principe de la métaphore ne répond pas vraiment à ce principe car l'élément central de la négociation est la table, puis les acteurs et enfin les propositions. Les négociations doivent être centrées sur les propositions et non pas sur les acteurs.

Une solution aurait pu être de mélanger à la fois la métaphore et le graphe d'enchaînement des états, mais cela aurait fortement alourdi l'écran.

Outre les requêtes directes, ARBAS96 dispose aussi de **requêtes indirectes**. Ces requêtes sont destinées à offrir aux utilisateurs du système les moyens d'accéder aux informations qui ne sont pas accessibles par les requêtes directes.

D'abord, nous tenons à préciser qu'ARBAS96 ne peut pas effectuer toutes les requêtes possibles et imaginables. Considérons la requête suivante *Quel est le but qui a été émis dans un problème traitant de commerce, qui a impliqué le secrétaire d'un autre problème qui s'est terminé par l'émission d'une proposition dont trois arguments sont favorables ! ! !*. Cette requête ne peut pas être directement interprétée par ARBAS96. Si une telle requête s'avérait nécessaire à l'avenir, nous proposerions l'utilisation d'un langage de requêtes.



Une idée que nous avons étudiée pour effectuer les requêtes indirectes est d'utiliser le schéma Entité-Association comme élément central de ces requêtes (Petout et Pigneur (1989), Bodart et al. (1994)). Le projet consiste à composer les requêtes au fur et à mesure en posant des valeurs de recherche dans les différents types d'entités et types d'associations du schéma. Les composants du schéma Entité-Association s'allument ou s'éteignent en fonction des éléments présents dans la requête. Cette représentation de la requête peut être comparée à un circuit électrique. Nous n'avons pas davantage développé cette idée pour deux principales raisons. D'abord, il nous a semblé que le schéma Entité-Association d'ARBAS96 était trop complexe, trop chargé pour permettre une interprétation aisée par les utilisateurs. Ensuite, nous pensons que cette interprétation n'est pas nécessairement facile pour un utilisateur étranger à ce genre de schéma et que cette difficulté est accentuée dans le cas de requêtes non élémentaires.

Finalement, le moyen utilisé pour supporter les requêtes indirectes, dans ARBAS96, est le formulaire multi-feuilles. Un formulaire multi-feuilles est composé de différentes feuilles qui comprennent les critères de recherche et d'une liste reprenant tous les éléments qui satisfont à ces critères de recherche. La requête se construit en passant de feuille en feuille. Le résultat de la recherche est affiché dans la liste qui évolue au fur et à mesure que la requête est complétée. Ce principe permet à l'utilisateur d'orienter les recherches dans la direction qui lui correspond le mieux.

A titre d'exemple, nous montrons de quelle manière une requête portant sur des problèmes peut être formulée. Les critères de recherche d'un problème ont été présentés précédemment.

**Query on problems**

Objective  
Termination  
Subject  
Evaluation  
**Date**  
Actor  
Event

**Date**

From : 28/06/95

To : 15/10/95

**Problems**

Problème1  
Problème2  
Problème4  
Problème6

OK Cancel

Figure II.13. : Liste des problèmes qui répondent au critère de date.

Considérons qu'ARBAS dispose de sept problèmes dans sa mémoire. En entrant le critère de date, délimitant un intervalle de temps durant lequel un problème est apparu, le formulaire présente une liste de problèmes qui répondent à ce critère (les problèmes 3,5 et 7 ne sont plus repris dans la liste) (Figure II.13.).

En ne conservant que les problèmes qui traitent de commerce international, par exemple, cette liste peut encore diminuer (Figure II.14.).



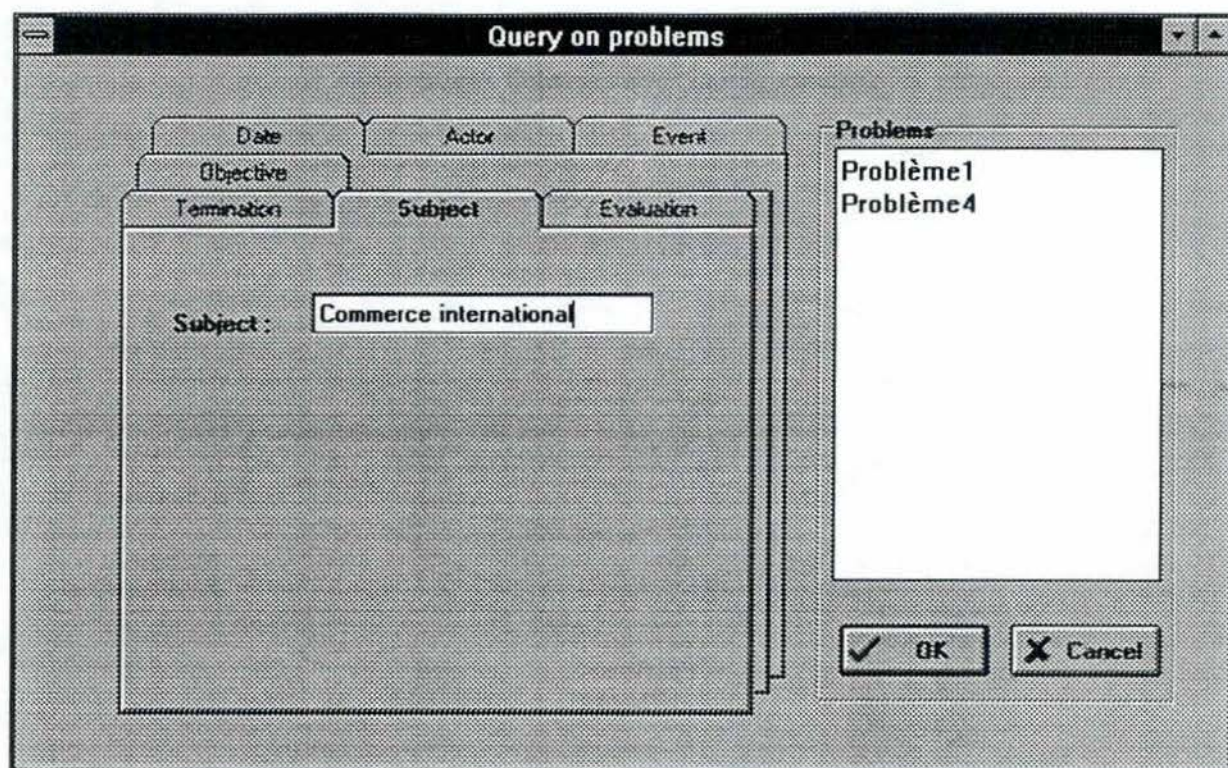


Figure II.14. : Liste des problèmes qui répondent aux critères de date et de sujet.

La liste de problèmes est évolutive. Chaque fois que l'utilisateur entre des valeurs de recherche dans une nouvelle feuille, la liste des problèmes diminue (ou stagne si le critère de recherche n'apporte rien de nouveau). Les problèmes répertoriés répondent à tous les critères instanciés dans les différentes feuilles du formulaire.

### II.5.7. Implémentation d'ARBAS96

Nous avons implémenté un prototype d'ARBAS96 dans un environnement Windows. Le langage de programmation que nous avons utilisé est le langage Delphi (Spoljar (1995)). Notre objectif était de montrer la faisabilité d'ARBAS96. Delphi nous a permis de définir rapidement la base de données, et surtout les écrans de l'application. Le prototype que nous avons implémenté ne contient pas l'ensemble des fonctionnalités décrites dans ce mémoire, mais il permet de visualiser le fonctionnement d'une négociation sous ARBAS96, c'est-à-dire la façon dont les acteurs définissent ou consultent un problème, émettent des propositions ou des arguments, etc... .

Nous présentons maintenant l'architecture d'ARBAS96 (Figure II.15.). Celle-ci est découpée en trois couches. Le niveau 3 contient le programme général et les différents modules qui reprennent les fonctionnalités du logiciel. Le niveau 2 s'occupe des interfaces

d'entrées/sorties. A ce niveau se retrouvent le module qui gère les accès à la BD. Le niveau 1 concerne les données globales utilisées par l'application.

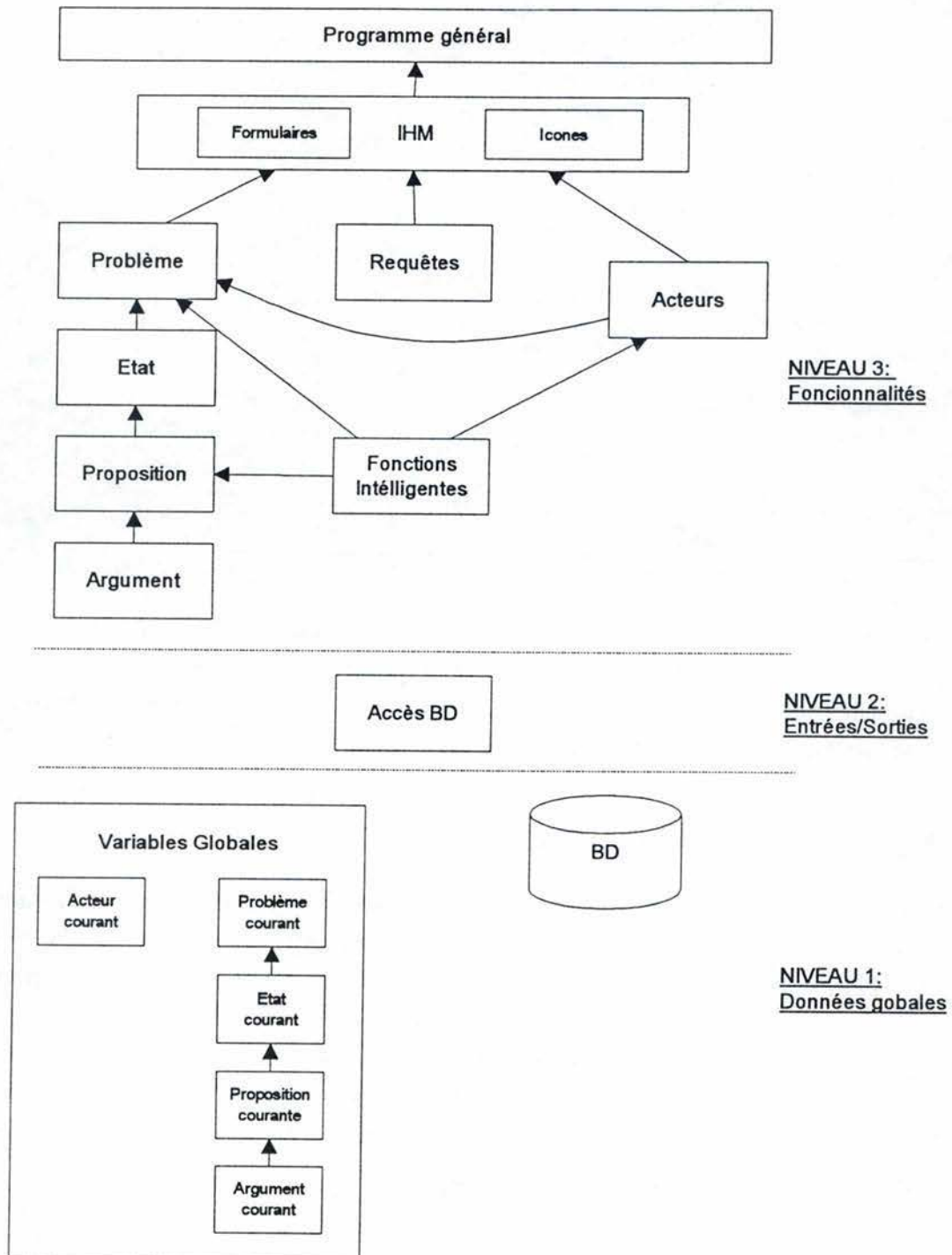


Figure II.15. : Architecture d'ARBAS96



Nous offrons, ci-après, un aperçu du contenu de ces modules:

### **Niveau 3**

#### Module Programme général:

Unité principale de l'application, fait les appels vers les modules contenant les fonctionnalités.

#### Module IHM:

Ce module objet gère les formulaires et les icônes présents dans l'application. Dans une programmation événementielle, c'est le module IHM qui fait appel aux autres modules.

#### Module Problème:

Module objet qui gère tout ce qui concerne un problème. Il contient un ensemble de fonctions de création, modification, suppression de problèmes et de leurs caractéristiques (Titre, Enoncé, Secrétaire, ...)

#### Module Requête:

Module fonctionnel qui offre une série de requêtes afin d'interroger le SI. Ces requêtes permettent aux acteurs d'obtenir des renseignements sur les négociations passées et actuelles.

#### Module Acteur:

Module objet qui s'occupe de la gestion des acteurs. Outre les traditionnelles fonctions de création/suppression, ce module offre la possibilité de consulter un fichier des acteurs pour mieux connaître leur profil.

#### Module Etat:

Module objet qui s'occupe de la création et de la consultation des états d'une négociation.

#### Module Proposition:

Module objet qui gère les propositions émises au cours d'une négociation. Ce module permet la création, modification, suppression de propositions et de leurs caractéristiques.

#### Module Argument:

Module objet qui gère les arguments émis à propos d'une proposition au cours d'une négociation. Ce module permet la création, modification, suppression d'arguments et de leurs caractéristiques.

#### Module Fonctions Intelligentes:

Module fonctionnel reprenant un ensemble de fonctions qui rendent le système actif. Il s'agit des fonctions qui interviennent dans la définition du problème. Ce module offre aussi les fonctions qui analysent le comportement des acteurs et suggèrent des propositions de solutions aux conflits.

#### **Niveau 2**

#### Module Accès BD:

Ce module fonctionnel assume le rôle d'intermédiaire entre l'application et sa base de données.

#### **Niveau 1**

#### Module Variables Globales:

Ce module contient un ensemble de modules objets liés à la gestion des variables globales de l'application : le problème courant, l'état courant, la proposition courant, l'argument courant ainsi que l'acteur courant.

#### La Base de données:

La base de données contient toutes les informations sur toutes les négociations et les acteurs.

### ***II.5.8. Etude de cas : Softwood Lumber***

#### **Présentation du cas**

Le cas Softwood-Lumber est un conflit entre les Etats-Unis et le Canada qui a débuté en Mai 1986 et s'est terminé le 30 décembre 1986. Ce cas, développé dans la littérature



(Kersten (1995)), a servi d'exemple pour la première version d'ARBAS (Binbasioglu et al. (1995)). Nous décrivons, ci-après, la situation de crise qui s'est présentée entre le Canada et les Etats-Unis.

En 1986, le Canada exportait des quantités énormes de bois vers les Etats-Unis alors qu'au même moment les Etats-Unis subissaient des pertes importantes sur ce même marché. Selon les Américains, la position forte du Canada sur le marché du bois (30 %) était la cause de leurs pertes. En fait, à cette époque, le Canada bénéficiait de faibles taxes d'abattage, ce qui, selon les Américains, était la source de cette grosse part de marché.

Les Américains se devaient de réagir et c'est ce qu'ils ont fait. Le 19 mai 1986, les Américains ont formé le American Coalition for Fair Trade (CFT), une coalition pour la défense d'un marché équitable. Cette coalition a envoyé une pétition au gouvernement américain (Département du commerce) et au U.S. International Trade Commission (ITC) pour régler le problème des exports canadiens.

Dans sa pétition, le CFT réclamait une taxe de 27 % sur les imports canadiens afin que le marché devienne plus équitable.

Le 26 juin 1986, ITC conclut, sur base d'une enquête, que le Canada portait, en effet, préjudice à l'industrie du bois américaine. Suite à cela, le Département américain du commerce (DOC) a décidé d'imposer une taxe d'importation de 15% à dater du 1 janvier 1987.

Afin de contrer cette décision, le Canada décida le 30 novembre 1986 d'augmenter ses taxes d'abattage. Cette solution n'arrange pas spécialement les affaires des bûcherons canadiens, mais au moins, les taxes sont perçues par le Canada et pas par les Etats-Unis.

En réaction à cette nouvelle, le CFT a retiré sa pétition ce qui a mis fin au conflit entre le Canada et les Etats-Unis.

### **Préparation du système**

De manière à pouvoir effectuer de l'apprentissage appliqué à la négociation, nous avons introduit dans la mémoire organisationnelle trois cas de négociations fictives parfois proches de cas réels. Ces informations seront utiles pour la définition du problème Softwood Lumber et pour retrouver des propositions et arguments de problèmes passés.

#### **a. Vache folle.**

Ce problème est lié aux exportations de viande depuis la Grande-Bretagne vers la France. Le bétail anglais souffre d'une maladie, appelée maladie de la vache folle, qui selon certains experts est transmissible à l'homme avec des conséquences fatales. Les principaux acteurs impliqués sont la France, via son ministère de la santé et le ministère anglais du commerce extérieur. Le secrétaire de la négociation est la Cour Européenne de Justice. La France émet comme proposition d'interdire les importations de viandes anglaises. Face à cela, les Anglais prétendent à l'aide d'une série d'arguments (appuis scientifiques) que la maladie n'est pas transmissible à l'homme. Ces arguments ne convainquent pas le département français de la santé qui ne change pas sa position.

Les Anglais doivent trouver une solution et décident d'abattre 11 millions de bêtes et de mettre les carcasses brûlées dans la mer du Nord. Le Commandant Cousteau s'oppose violemment à cette proposition car la maladie peut être transmise aux poissons. Les végétariens contestent aussi la décision anglaise et lancent une pétition. La situation est bloquée à ce point, pour l'instant.

Le problème est enregistré, sous le titre vache folle, avec les acteurs présentés ci-dessus. Les buts des intéressés sont la santé (France), le profit (UK) et le respect des animaux (Cousteau et Association de végétariens). Les mots-clés sont : commerce, viande, vache folle, santé, international.

#### **b. Commerce déloyal entre la Chine et les Etats-Unis.**

Ce problème de commerce déloyal concerne les exportations de pétrole depuis les Etats-Unis vers la Chine. La compagnie Rallas se plaint de ne pouvoir obtenir plus de 10% du marché chinois et réclame un marché plus équitable. Le titre donné au problème est Softoil et il implique le département du commerce américain, le gouvernement chinois et la compagnie pétrolière américaine Rallas. Le secrétaire du problème est la Commission Internationale pour l'Honnêteté des Marchés (CIHM). Les différents mots-clés posés sur ce problème de négociation sont : International, Commerce, Equité, Pétrole. Le but de la Chine est de favoriser ses propres sociétés, et celui de Rallas est d'avoir droit à plus d'équité.

Après de longues discussions que nous ne développerons pas ici, les Chinois sont d'accord de monter la limitation du marché à 20 %. En échange de quoi, les Américains autorisent les Chinois à importer davantage d'antiquités sur le marché américain.

#### **c. Amazonie**



Ce problème est lié à l'exploitation d'une partie de la forêt amazonienne. Deux sociétés se partagent une zone de la forêt amazonienne : la société Raoni et la société McDonald. La société Raoni désire exploiter la zone en construisant des magasins de souvenirs pour touristes (e.g. plateau à mettre en bouche). Cette position ne satisfait pas McDonald qui désire raser la zone pour y placer du bétail qui finira en hamburgers. Les deux parties ont émis tour à tour des propositions mais elles n'arrivent pas à s'accorder. La négociation a été abandonnée car trop de temps s'est écoulé. ARBAS96 a enregistré cette négociation dans les négociations défectueuses. Le titre du problème est Amazonie et les buts des acteurs est le profit. Les mots-clés sont viande, commerce, écologie.

### Softwood Lumber par ARBAS96

La première étape qui est proposée par ARBAS96 est l'enregistrement des caractéristiques principales du problème qui vient d'apparaître : le titre du problème, la date du problème (semi-automatique), l'énoncé du problème, les mots-clés du problème, une première liste d'acteurs et les événements déclencheurs. Pour enregistrer un problème avec ARBAS, il faut ouvrir un nouveau problème et remplir les champs tel qu'il est demandé (voir Figure II.16.). Comme indiqué sur la figure, nous avons opté pour les mots-clés suivants : international, commerce, bois, équité.

The screenshot shows a window titled "New Problem". It contains the following fields and text:

- Title :** Softwood Lumber
- Date :** 19/06/86
- Description :** L'industrie américaine du bois se plaint du commerce non équitable des Canadiens qui bénéficient d'une taxe d'abattage inférieure.
- Key words :** International, Commerce, Bois, Equité
- Events :** Part de marché > 30 %

At the bottom of the window are three buttons: Ok, Cancel, and Help.

Figure II.16. : Capture des caractéristiques de base du problème



Dès que l'utilisateur du système, qui n'est pas nécessairement le secrétaire ou un participant de la négociation, a entré ces caractéristiques, ARBAS96 fouille sa mémoire à la recherche d'un acteur susceptible d'assumer la fonction de secrétaire. Le premier secrétaire proposé est la Commission Internationale pour l'Honnêteté sur les Marchés (CIHM) (voir Figure II.17.). Ce secrétaire avait déjà participé à une négociation internationale entre les Etats-Unis et la Chine (cas 2 dans la mémoire). Le CIHM accepte, à nouveau, la fonction de secrétaire et prépare la suite de la négociation.

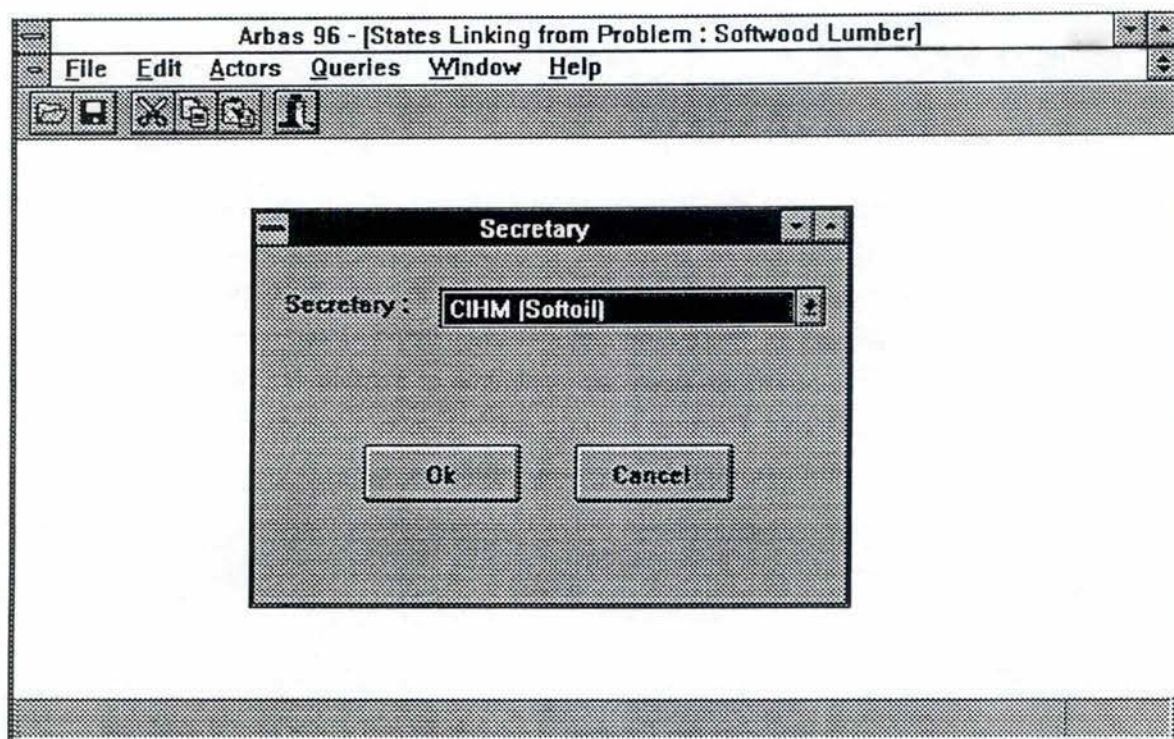


Figure II.17. : Sélection d'un secrétaire

Pour l'introduction des acteurs de la négociation, ARBAS96 relève des spécialistes dans le genre de cette négociation. Dans le cas qui nous occupe, il s'agit des acteurs du cas 2 (conflit Etats-Unis/Chine), puis les acteurs du conflit de vache folle (cas1) et enfin les acteurs du cas Amazonie (voir Figure II.18.). Ces spécialistes peuvent aider le secrétaire pour l'énumération des acteurs de la négociation Softwood.

D'abord le secrétaire constate qu'une plainte a été introduite par le CFT. Le CFT n'a jamais été acteur dans une négociation précédente et, pour cause, puisque ce comité a été créé, intentionnellement, pour réagir face à la forte position du Canada sur le marché. Le



secrétaire ajoute donc le CFT parmi les acteurs. Il complète aussi la liste des acteurs par le Canada en regard aux informations détenues dans la plainte du CFT.

Le secrétaire insère aussi le département de commerce américain qui a déjà été acteur dans une négociation de commerce international.

ARBAS96 vérifie si les acteurs n'ont pas déjà constitué un "*cocktail perdant*" dans le passé. Ce n'est pas le cas sinon ARBAS96 aurait mentionné ce fait au secrétaire.

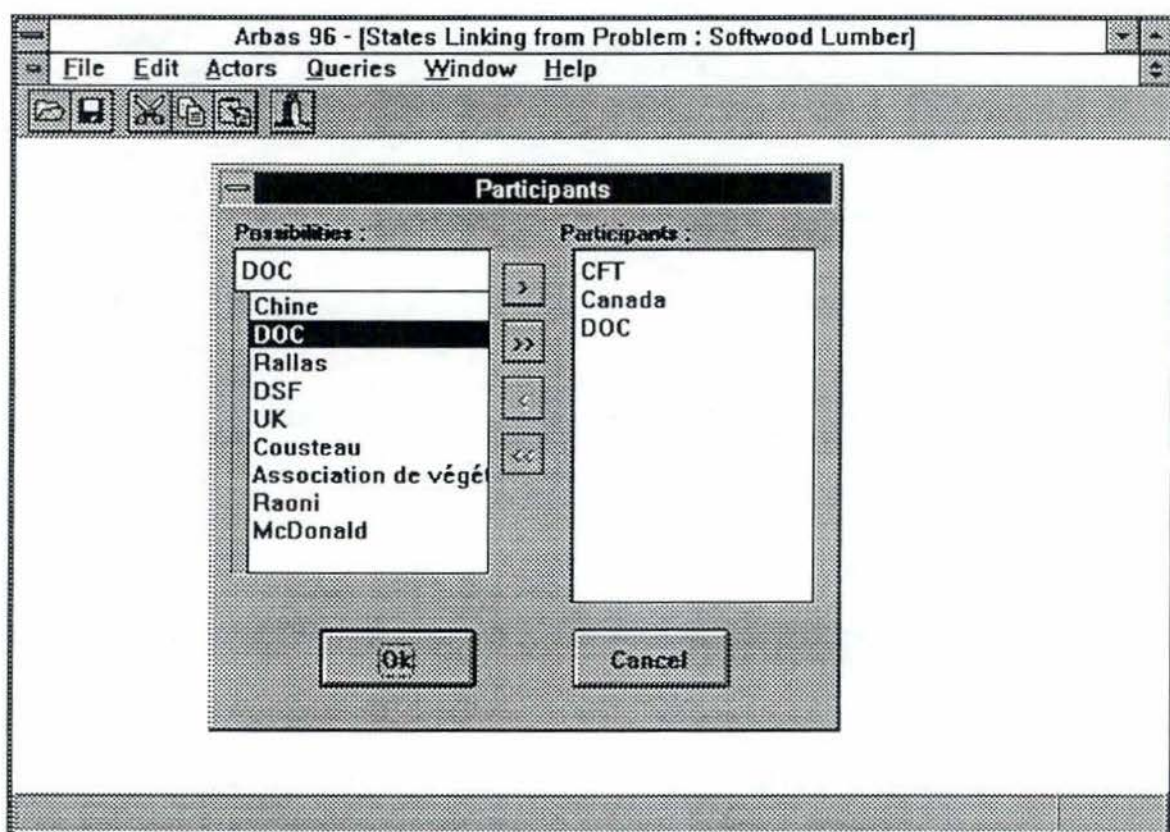


Figure II.18. : Les participants d'une négociation

Les acteurs sont ensuite invités à émettre des buts. Le but du Canada est de ne pas payer de taxes aux Etats-Unis (critère : taxe à payer aux Etats-Unis; valeur : 0) et le but des Etats-Unis est de rendre le marché équitable (critère : équité du marché; valeur : indéterminée). Le protocole choisi est l'intervention libre, sans anonymat.

Le problème est complètement défini et il faut maintenant proposer une solution à ce problème. Le système va suggérer une proposition de solution. Le cas 2 ressemble le plus au cas Softwood Lumber puisque 4 mots-clés concordent et un des buts des acteurs est



identique, à savoir : l'équité. Le système propose une solution au conflit : " Limitation légale des importations à 20 % du marché" (Figure II.19.).

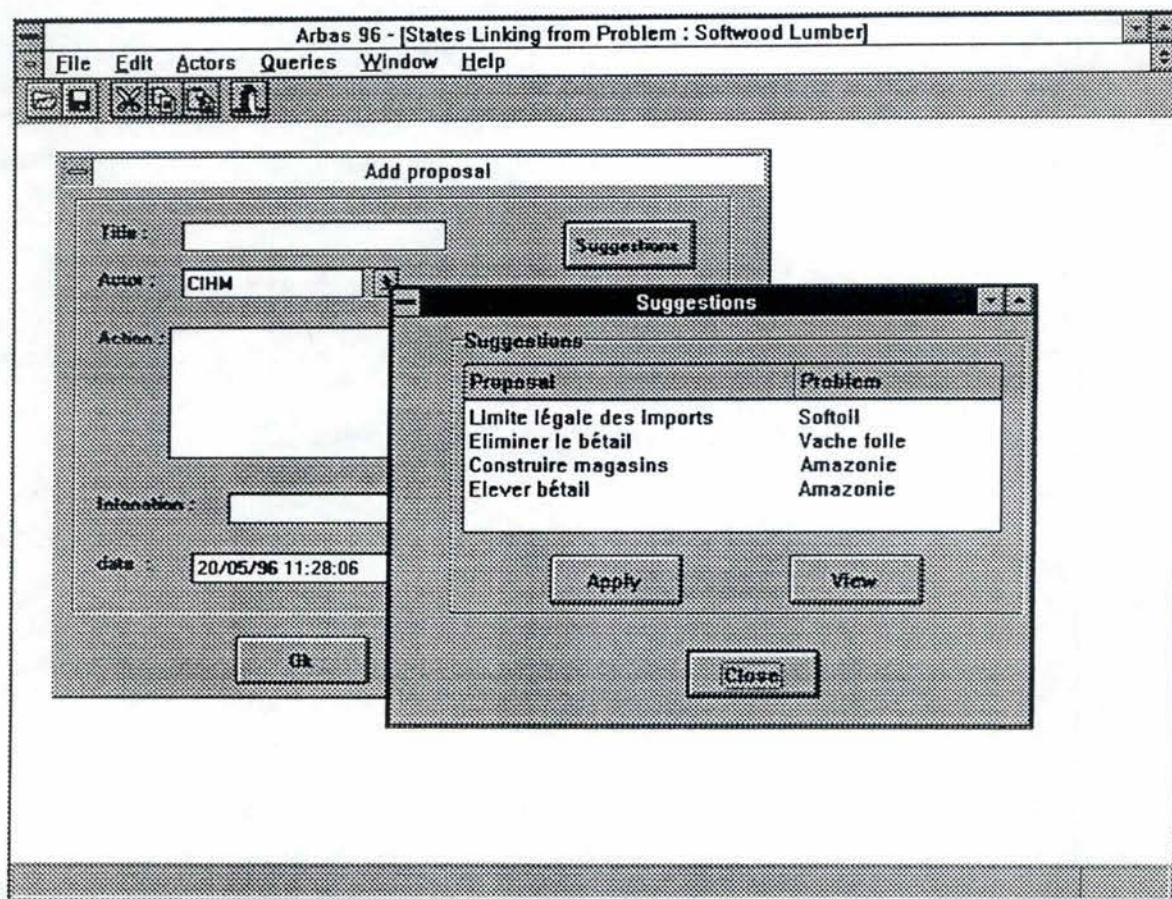


Figure II.19. : Emission d'une proposition et détails des suggestions

Le secrétaire analyse cette proposition mais constate que cette solution n'est pas applicable au cas Softwood Lumber, car des décrets spéciaux sont signés avec le Canada, lui autorisant à occuper le marché américain sans limites de parts de marché. Le secrétaire n'émet aucune proposition et attend que l'un des acteurs le fasse. Le CFT agit en émettant une proposition de solution au conflit : " *taxe de 27 % sur les imports canadiens*". Pour argumenter sa proposition, le CFT soulève le fait que le commerce est déloyal.

Cette proposition est rendue publique par ARBAS. L'émission d'une nouvelle proposition provoque la création d'un nouvel état et en cliquant sur la proposition, l'utilisateur peut obtenir, dans une fenêtre, les détails de cette proposition (Figure II.20.).



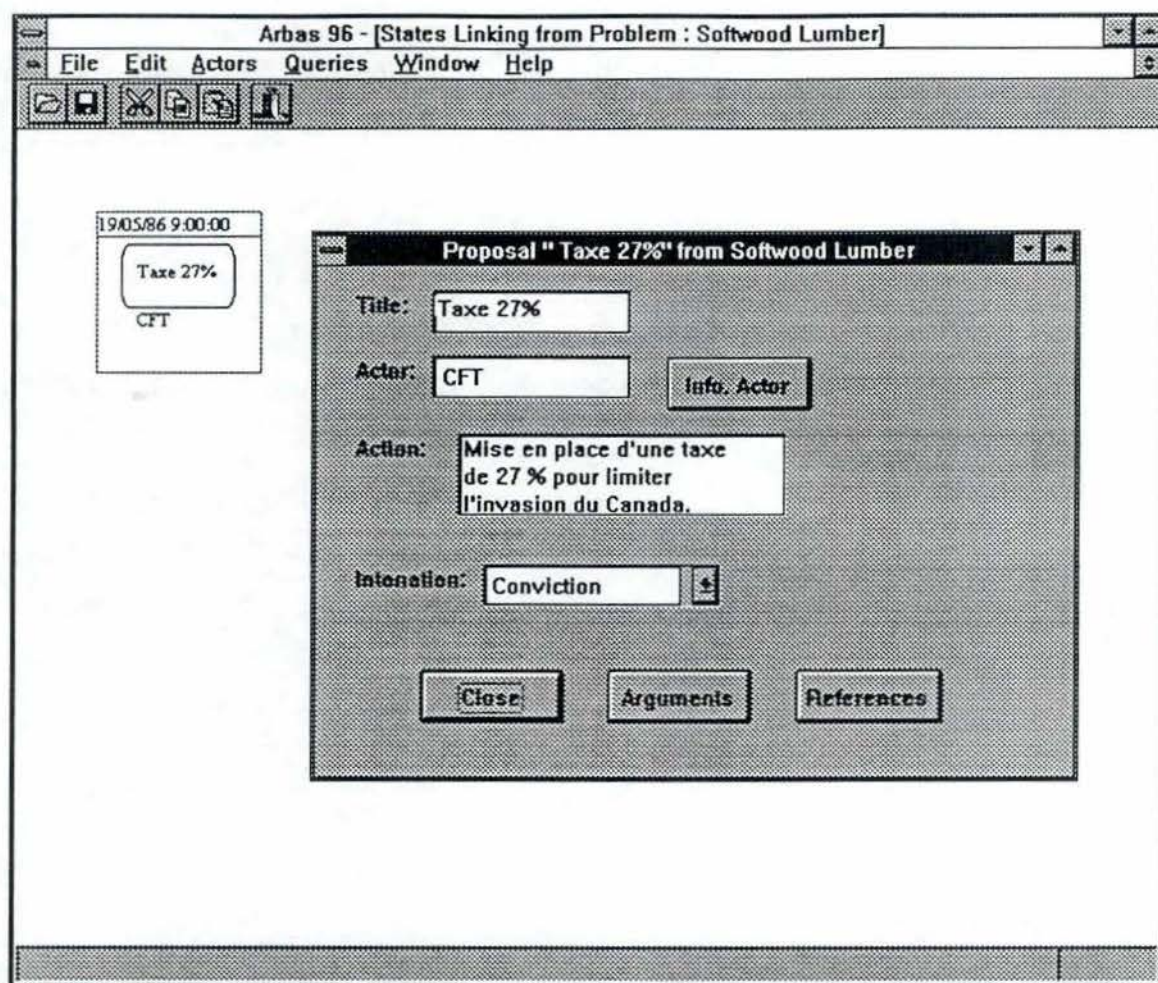


Figure II.20. : Etat constitué de la première proposition et le détail de celle-ci

Les réactions ne se font pas attendre. L'ITC fait son entrée dans la négociation en émettant un argument supportant l'argument du CFT. L'ITC reconnaît que le Canada porte préjudice aux Etats-Unis. Le Canada émet un argument défavorable à l'argument amené par l'ITC. Nous rappelons que les arguments positifs sont des rectangles, les ellipses sont des arguments négatifs et les triangles sont des arguments neutres. Tout comme pour les propositions le détail de chaque argument s'obtient en cliquant dessus (Figure II.21.).



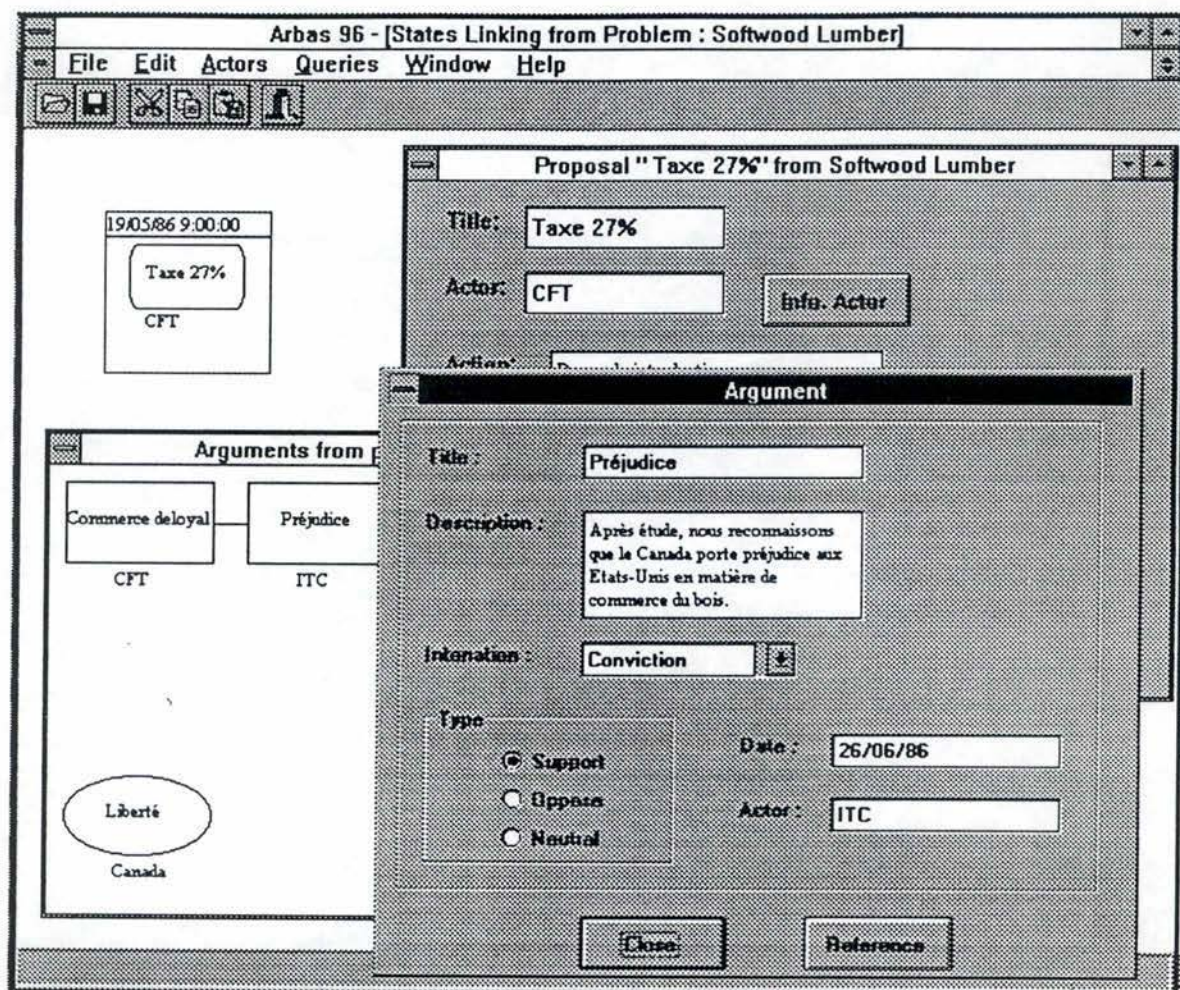


Figure II.21.: Arguments d'une proposition

Le Département du Commerce, sur base de la proposition du CFT et se servant du rapport de l'ITC pour références, émet une nouvelle proposition : "Taxe de 15 % sur les imports canadiens". L'apparition d'une nouvelle proposition entraîne la création d'un nouvel état dans lequel cohabitent la proposition du CFT et celle du DOC. Suite à la proposition du DOC, le CFT retire sa proposition car il est satisfait (figure II.22.).



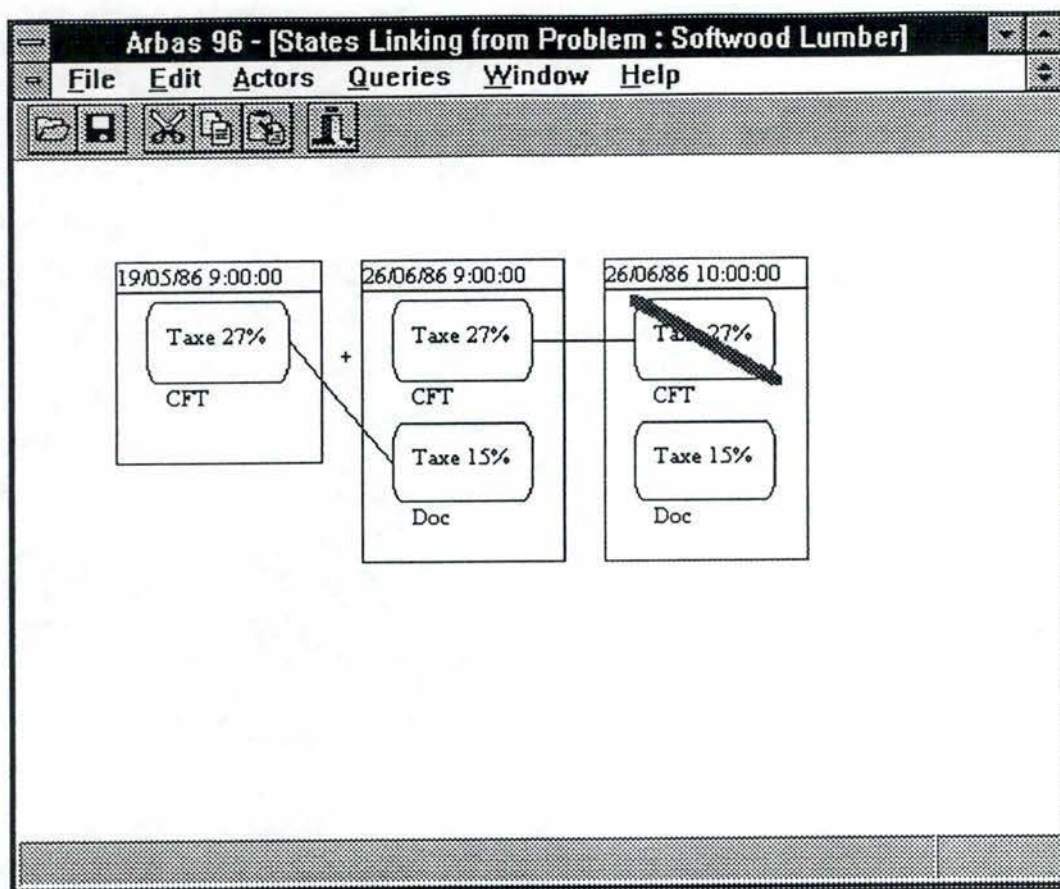


Figure II.22. : Le DOC propose une taxe de 15% et le CFT retire sa proposition

Afin de contrer la proposition du DOC, le Canada doit trouver une solution. Il consulte ARBAS96 pour étudier quels sont les problèmes qui traitent de commerce international et où les buts émis sont liés au profit. En remplissant un formulaire de requête, un seul problème répond aux caractéristiques recherchées (cas de vache folle) (voir Figure II.23.).

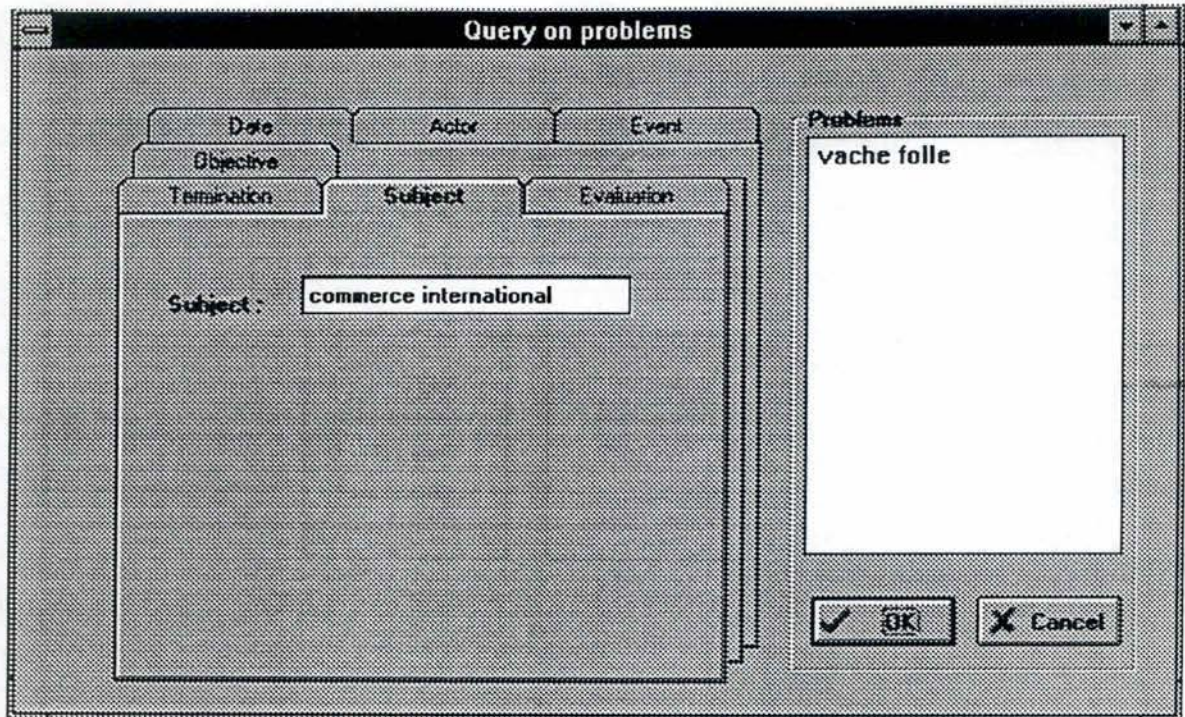


Figure II.23.: Requête de recherche de problèmes

Le Canada ouvre le problème qu'il vient de trouver et analyse la négociation. Celle-ci n'est pas encore terminée. Il constate que la Grande-Bretagne veut abattre son bétail pour limiter ses exportations de viande contaminée (Figure II.24.).



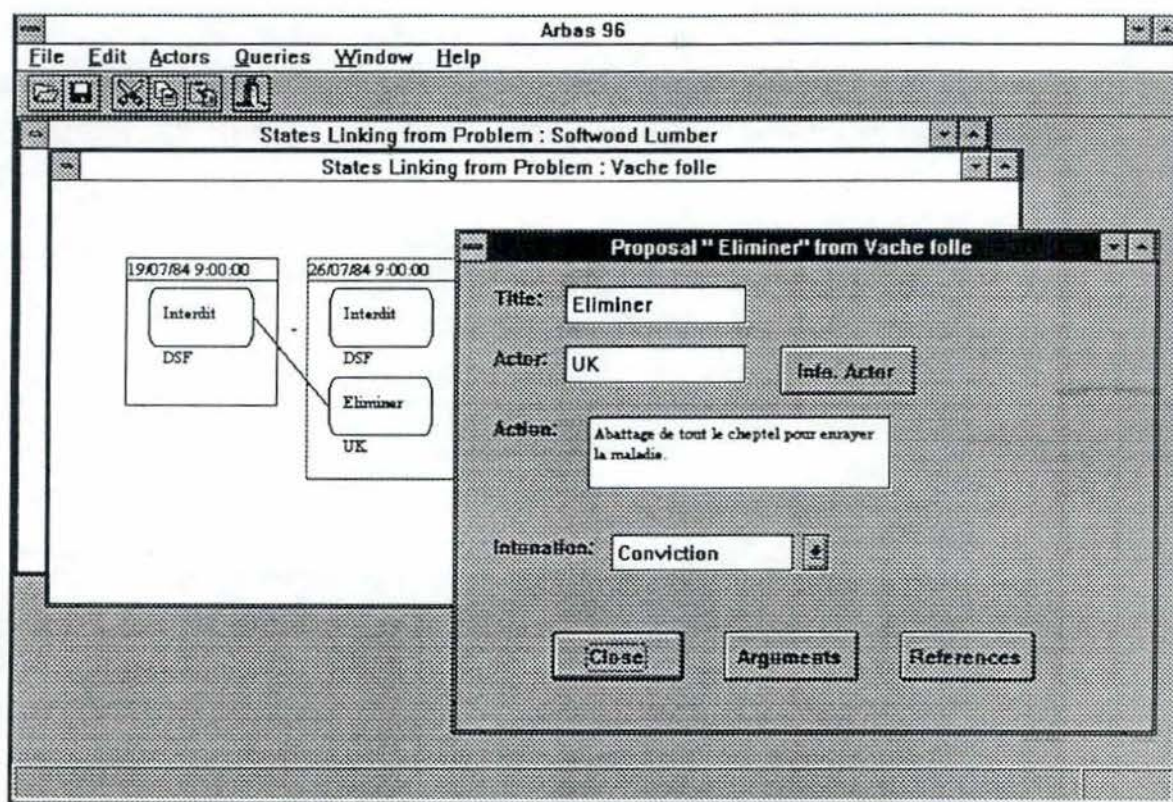


Figure II.24. : Analyse du cas de vache folle

Le Canada est inspiré par cette idée d'auto-sanction et décide d'augmenter ses propres taxes d'abattage. Cette nouvelle proposition entraîne la création d'un nouvel état (voir Figure II.25.).

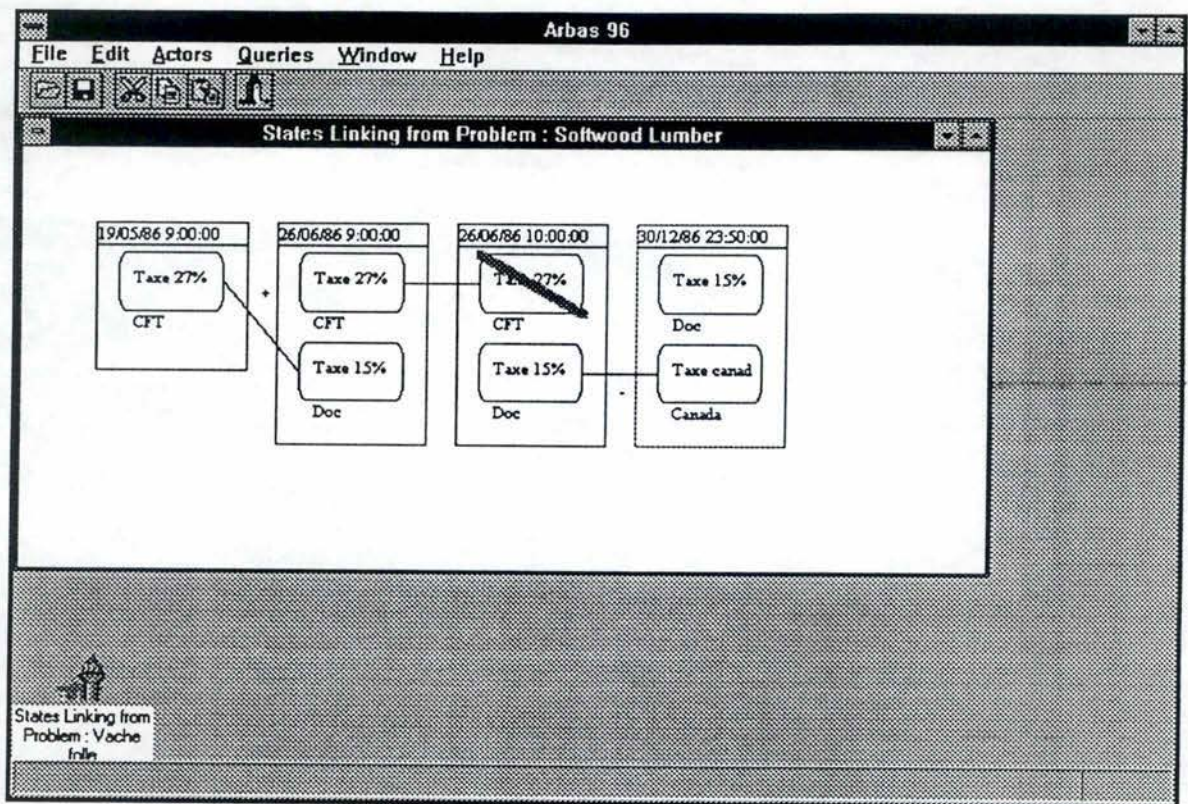


Figure II.25. : Emission de la proposition du Canada

Enfin, face à la proposition du Canada, le DOC décide de retirer sa proposition, ce qui met fin au conflit et crée l'ultime état (voir Figure II.26.).



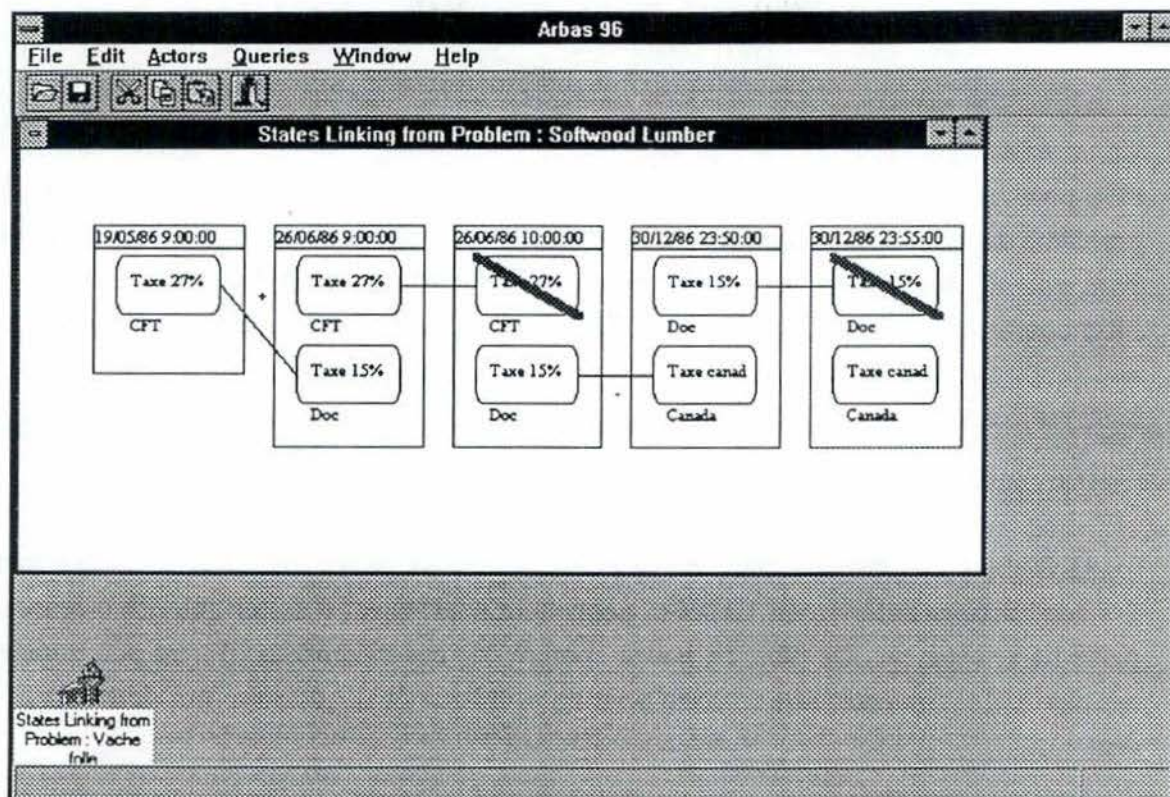


Figure II.26. : Evolution du problème jusqu'à l'ultime état

Le conflit s'est achevé avec succès et donne lieu à un accord entre le Canada et les Etats-Unis.

### II.5.9. ARBAS96 est un SISMO

Dans cette section, nous montrons de quelle manière ARBAS96 répond réellement aux exigences des SISMOs développées dans le modèle de Stein et Zwass (1995).

#### Intégration dans l'espace

ARBAS96 offre aux acteurs une plate-forme de travail virtuelle qui leur permet de s'intégrer à une négociation sans devoir se déplacer physiquement à la table de négociation. La plate-forme répond directement aux exigences de l'intégration dans l'espace. De plus, ARBAS96 enregistre des informations sur l'intégration spatiale des individus au cours du temps (e.g. à tel moment, telle personne occupait telle fonction dans tel département). Cette mémorisation permet de définir quelles sont (ou quelles étaient) les responsabilités de chacun.

### **Intégration dans le temps**

ARBAS96 mémorise des négociations passées et les rend accessibles aux utilisateurs via un ensemble de requêtes génériques. ARBAS96 se sert aussi de l'intégration temporelle pour agir de manière active dans une négociation. Lorsqu'un utilisateur ouvre un problème dans ARBAS96, il peut y *voyager*. Les utilisateurs ont la possibilité de changer l'état courant d'une négociation et d'ainsi revenir dans le passé. Cette modification de l'état courant est un bon moyen pour l'utilisateur de comprendre le contexte dans lequel une proposition, par exemple, a été émise. Les éléments de contexte sont les acteurs, leurs buts et critères, le protocole, l'état des propositions et des arguments.

### **Adaptation**

ARBAS96 mémorise le passé ce qui permet aux acteurs de tirer des enseignements des négociations antérieures. Celles-ci permettent notamment d'éviter que des erreurs du passé ne se répètent. De plus, en tenant compte des cas mémorisés, il n'est pas toujours nécessaire de réinventer une solution pour un problème de négociation qui s'est déjà produit.

Le fait d'enregistrer tous les cas de négociation offre aussi un moyen pour les nouveaux venus de s'adapter rapidement aux problèmes de négociation. En consultant les problèmes, ils peuvent vite comprendre les erreurs ou les succès qui ont traversé l'histoire de l'organisation.

Un ensemble de boucles se retrouvent dans les schémas de fonctionnement d'ARBAS96 (voir point II.5.5.). Elles montrent bien le déroulement évolutif et adaptatif des phases de la négociation. Chaque participant visualise les états consécutifs du problème de négociation et peut ainsi adapter son jugement et proposer des solutions plus compatibles avec celles des autres participants.

Des outils intelligents d'ARBAS96 offrent aux acteurs des analyses des problèmes passés. Celles-ci vont étendre leurs connaissances et les aider à émettre des propositions plus intéressantes. Enfin, ARBAS96 enregistre les événements qui sont à l'origine d'un conflit. Il est ainsi possible de comprendre de quelle manière l'organisation s'adapte face à un événement qui se produit dans son environnement. De même, à une proposition ou un argument peuvent être rattachés des objets de référence externes à l'organisation, ce qui crée un lien entre l'organisation et son environnement.

### **Réalisation de buts**



Dans ARBAS96, il est demandé aux acteurs de motiver, à l'aide de buts et de critères, leur participation à la négociation. Les buts peuvent être attachés à des objectifs organisationnels. L'existence de ce lien offre un moyen de percevoir quels sont les objectifs de l'organisation et quels sont les problèmes qui y sont liés. L'évaluation du degré de réalisation d'un objectif organisationnel peut être inférée à partir des différents problèmes de négociation.

Le fait de poser un lien entre les buts individuels et les objectifs organisationnels pousse les acteurs à diriger la négociation selon une orientation qui sera bénéfique pour l'organisation.

### **Maintenance des procédures de travail**

ARBAS96 conserve les négociations auxquelles des acteurs de l'organisation ont été impliqués. Ces informations peuvent être très utiles pour un service de ressources humaines. En effet, à l'aide d'ARBAS96, il est possible de comprendre le comportement des individus et de l'évaluer. Certaines personnes sont plus aptes à travailler dans un certain type de négociation, tandis que d'autres se sont toujours montrées inefficaces. ARBAS96 contient aussi un outil qui va analyser la bonne composition d'une liste d'acteurs. Détecter, par exemple, que les négociations passées impliquant les acteurs ne se sont pas trop mal passées.

Les propositions, les arguments, les feed-backs, les buts et les critères permettent de mieux connaître l'efficacité des acteurs et également leur comportement en groupe.

En conclusion, nous pouvons prétendre qu'ARBAS96 répond aux exigences du modèle de Stein et Zwass (1995). Il est maintenant intéressant de vérifier qu'ARBAS96 dispose bien des qualités qu'un SISMO devrait pouvoir présenter (voir point I.8.3.) : partageable, transportable, sûr, précis, à jour et pertinent (Watson (1996)).

Comme nous l'avons précisé plus haut, la plate-forme offerte par ARBAS96 permet à différents utilisateurs d'accéder à des données communes (partageable) et distribuées (transportable). L'aspect sécurité des données est garanti par l'utilisation d'un mot de passe pour entrer dans le système. Davantage de précautions en matière de sécurité pourraient évidemment être prises : protection physique du système, copies de sécurité, ... . Ces éléments n'ont pas été développés dans la mesure où ils ne sont pas spécifiques au monde de la négociation, mais concernent toutes les données d'une organisation.

La précision des informations des données est importante. ARBAS96 offre via les propositions et les arguments des moyens d'expression efficaces. L'utilisation d'objets de référence peut aussi augmenter la précision des propositions et arguments.

La maintenance du système ARBAS96 offre la garantie que les données qu'il contient sont à jour. ARBAS96 permet, par exemple, de supprimer des problèmes, des propositions ou des arguments obsolètes.

La pertinence des données est assurée par le fait qu'ARBAS96 ne mémorise que les éléments qui sont directement liés à son activité : la résolution de processus de négociation.

Dès lors, puisque nous venons de montrer qu'ARBAS96 respectait bien les exigences du modèle de Stein et Zwass (1995) et qu'il présentait les différentes qualités qu'un SISMO doit pouvoir offrir, nous pouvons donc prétendre qu'ARBAS96 est bien un SISMO. Il se différencie des exemples de SISMOs que nous avons présentés par le fait qu'il capture des éléments de chaque sous-système du modèle de Stein et Zwass (1995), ce qui le rend plus efficace d'un point de vue organisationnel.

#### ***II.5.10. Limites d'ARBAS96 et vers ARBAS97***

ARBAS96 a été conçu pour essayer de combler les manques de la précédente version d'ARBAS ainsi que pour supporter d'une meilleure façon la mémoire organisationnelle liée à la négociation. Cependant, nous avons remarqué qu'un certain nombre d'inconvénients restaient à résoudre.

Tout d'abord, le problème doit être défini par un responsable, le secrétaire. Une possibilité est que le secrétaire soit l'un des acteurs de la négociation. Mais que se passe-t-il s'il désire quitter la négociation? S'il agit de façon incorrecte (refuser l'accès à un acteur)? La solution à ce problème serait de choisir une personne neutre qui ne participe pas à la discussion mais cette personne ne serait-elle pas un facilitateur? Ce qui serait contraire à la philosophie d'ARBAS.

Une autre limite d'ARBAS96 concerne les propositions. Si une personne émet une proposition puis quitte la négociation, personne ne peut savoir que cette proposition n'a plus de raison d'être. Etant donné que seul l'auteur d'une proposition a le droit de la supprimer, celle-ci va persister et gêner le bon déroulement de la discussion. La réponse à



cette limite consiste à purger régulièrement les propositions. Soit par un tour de table soit en définissant une durée après laquelle une proposition doit être reconfirmée.

Ensuite, il y a une limite liée à la représentation de l'enchaînement des états. Lorsque la négociation se déroule sur une longue période et/ou comporte de nombreuses émissions de propositions, il est impossible de visualiser l'ensemble des états sur l'écran (il n'est pas assez large). De plus, si un problème génère un grand nombre de propositions par état, c'est-à-dire qu'à un moment donné, beaucoup d'acteurs proposent une action différente, alors l'état ne pourra pas être visualisé en entier (il n'est pas assez haut). Il faut dès lors permettre aux utilisateurs de naviguer le mieux possible au sein de l'interface à l'aide de barres de défilements et de requêtes de recherche d'états et de propositions.

Par rapport à l'apprentissage, nous pouvons aussi soulever un point de discussion. Comme, nous l'avons montré ARBAS96 permet aux utilisateurs de rechercher des informations mémorisées dans des négociations passées et de les réutiliser dans leurs propres négociations. ARBAS96 automatise une partie des recherches en proposant des acteurs ou des propositions qui ont trait à des problèmes similaires. Ceci peut être considéré comme insuffisant pour les partisans du case-based reasoning. En appliquant des théories d'intelligence artificielle et de case-based reasoning à ARBAS96, il y aurait moyen d'améliorer cet aspect. En case-based reasoning, il est nécessaire de délimiter un ensemble d'attributs sur lesquels les acteurs posent des valeurs dans leurs propositions. Toutefois, nous avons toujours considéré ARBAS96 comme un outil de support à la négociation et, de manière à ne pas entraver la liberté des utilisateurs du système, nous ne sommes pas convaincus qu'il faille aller plus loin dans l'introduction de processus intelligents car cette extension risque de diminuer la souplesse d'ARBAS96.

Toutefois, nous pensons qu'il serait intéressant d'améliorer le contrôle effectué par ARBAS96 en cours de négociation. Pour rappel, ARBAS vérifie que des propositions soient émises de manière régulière dans la négociation. Cette fonction pourrait être spécialisée au niveau des acteurs. Il s'agirait de *secouer* les acteurs qui ne participent pas assez à la négociation. La fonction que nous proposons photographierait, à certains moments, la situation de la négociation déterminée notamment par le comportement des acteurs.

Lorsque ARBAS96 suggère une liste de propositions de solution à un problème, l'acteur analyse cette liste et éventuellement le détail des propositions. Certaines solutions ne sont pas applicables et sont rejetées par cet acteur. Nous pensons qu'il peut être intéressant de mémoriser la condition de rejet d'une suggestion, dans le but d'acquérir plus d'informations sur une proposition.



Enfin, nous avons montré qu'ARBAS96 pouvait intégrer les informations dans l'espace. Cependant, nous devons préciser que cette intégration ne se fait que d'un point de vue logique car ARBAS96 ne supporte pas encore le partage des connaissances du point de vue technique (réseaux de communication, groupware, ....). Une future version d'ARBAS devra faire face à cette lacune.

## **II.6. Résumé**

---

Récemment, il a été constaté une prolifération d'applications et de systèmes *groupware*. Ces systèmes permettent principalement à des personnes de coopérer et de communiquer sur des actions communes. Parmi ces actions, nous relevons la négociation comme un processus impliquant plusieurs parties cherchant à atteindre une solution dans un conflit.

Dans cette partie, nous avons montré en quoi il est intéressant de mémoriser les processus de négociation et nous avons aussi souligné les avantages liés à l'utilisation de systèmes d'aide à la négociation.

Notre attention s'est ensuite portée sur ARBAS, un langage d'argumentation. Dans le cadre de la négociation, nous lui avons relevé un certain nombre de limites auxquelles nous avons apporté une première extension.

Nous avons aussi étendu ARBAS pour qu'il réponde davantage aux exigences des systèmes d'information supportant de la mémoire organisationnelle, toujours dans le cadre de la négociation. Cette extension a été accompagnée par l'introduction de nouveaux concepts et fonctions. Par la force des choses, le fonctionnement d'ARBAS étendu, nommé ARBAS96, a été amélioré. Etant donné la quantité d'informations présentes dans ARBAS, nous avons aussi voulu rendre ce système plus actif, plus intelligent au cours des négociations. Par exemple, ARBAS96 peut proposer des solutions ou émettre au secrétaire des messages d'avertissement pendant la négociation.

Dans cette partie, nous avons montré la force du langage d'argumentation ARBAS. Il est un moyen d'expression aisé pour les participants d'une négociation, il permet aussi de contrôler les étapes de la négociation et, enfin, il offre les moyens de récupérer toute sorte d'informations liées aux processus de négociation. De ces informations, il est aussi possible d'inférer le climat général d'une organisation ou d'évaluer les membres de cette organisation.



---

## Conclusion

Actuellement, les organisations sont amenées à mémoriser de plus en plus d'informations liées à leurs activités. L'ensemble de ces informations constituent ce qu'on appelle la mémoire de l'organisation, ou encore, la **mémoire organisationnelle**. La principale utilisation de celle-ci est l'apprentissage organisationnel qui consiste à se servir des expériences du passé pour améliorer celles du présent.

Dans le cadre de ce travail, nous nous sommes intéressés à la mémoire organisationnelle liée aux processus de négociation au cours desquels une organisation (ou une partie d'une organisation) est amenée à prendre des décisions. L'intérêt de notre travail réside dans le fait que, pour l'instant, les différents modèles qui ont été exposés par rapport à la mémoire organisationnelle (Walsh et Ungson (1991), Stein et Zwass (1995), Morrison et Weiser (1996)) ne définissent pas concrètement quelles sont les données qui doivent être présentes dans un système d'information supportant la mémoire organisationnelle (SISMO).

La base de notre étude est le langage d'argumentation ARBAS (*Action-Resource Based Argumentation Support*) (Binbasioglu et al. (1995) et Bui et al. (1995)). Nous avons étendu ce langage selon deux directions. D'abord, nous avons développé ce langage pour qu'il réponde plus aux besoins des processus de négociation. Nous y avons introduit les notions d'état de négociation, de protocole de négociation, d'anonymat, et d'arguments.

Nous avons ensuite étendu ARBAS de manière à ce qu'il soit conforme aux exigences des SISMOs. Cette extension a donné naissance à ARBAS96. Le fruit de notre étude permet de détailler le contenu d'un SISMO, en termes de concepts et de fonctionnalités, lorsque celui-ci est lié au processus de résolution de conflits.

L'extension que nous avons menée, nous a poussés à introduire de nouveaux concepts et de nouvelles fonctions qui nous ont principalement permis :

- **d'offrir un meilleur support informationnel aux processus de négociation.** Les processus de négociation regorgent d'informations qui peuvent servir de sources d'inspiration pour les futures négociations.

- **d'améliorer l'intelligence du système au cours des processus de négociation.** Etant donné les informations mémorisées dans ARBAS96, nous avons pu rendre le système plus actif au cours des négociations. Outre la proposition de solutions à des problèmes similaires, ARBAS96 vérifie aussi les différentes phases d'une négociation.
- **d'augmenter les informations disponibles dans une organisation.** En mémorisant des processus de négociation, ARBAS96 enregistre des informations par rapport à une partie de ses activités. Ces informations lui permettent, par exemple, de déduire les problèmes auxquels elle a été confrontée, le comportement des personnes qui ont été impliquées dans la résolution de ces problèmes, l'efficacité de ces personnes, etc... . Dans ces conditions, ARBAS96 ne doit pas exclusivement être considéré comme un outil d'aide à la négociation, mais peut aussi être vu comme un moyen pour récupérer, à un niveau organisationnel, des informations liées à l'histoire d'une organisation et de ses individus.

Finalement, de manière à montrer les possibilités d'ARBAS96 et la façon dont les utilisateurs l'utilisent, nous avons implémenté un prototype de ce langage d'argumentation.



---

## Bibliographie

Ackerman, M.S., 1994, "Definitional and Contextual Issues in Organizational and Group Memories", *Proceedings of the 27th Hawaii International Conference on Systems Science*, Janvier.

Ackerman, M.S. et Mandel, E., 1995, "Memory in the Small: An Application to Provide Task-Based Organizational Memory for a Scientific Community", *Proceedings of the 28th Hawaii International Conference on Systems Science*, Janvier.

Alter, S., 1992, *Information Systems : A Management Perspective*, Addison Wesley, Reading, Mass.

Argyris, C. et Schon, D.A., 1978, *Organizational learning: A theory of action perspective*, Reading, Addison-Wesley, Mass.

Balasubramanian, V., 1995, "Organizational Learning and Information Systems", Page Internet URL : <http://www.umd.edu/Business/AcademicDepts/IS/Learning/orgirn1.html>, Rutgers University, NJ.

Bannon, L. J. et Kuutti, K., 1996, "Shifting Perspectives on Organizational Memory: From Storage to Active Remembering", *Proceedings of the 29th Hawaii International Conference on Systems Science*, Janvier.

Binbasioglu, M., Bui, T. et Ma P-C., 1995, "An Action-Resource Langage for Argumentation : The Case of Softwood Lumber Negotiation", *Proceedings of the 28th Hawaii International Conference on Systems Science*, Janvier.

Bodart, F. et Pigneur, Y., 1989, *Conception assistée des systèmes d'information*, 2ème édition, Masson (eds), Paris.

Bodart, F., Hennebert, A-M, Leheureux, J-M, Provot, I., Vanderdonckt, J. et Zucchinetti, G., 1994, "Dimensions clé pour une méthodologie de développement d'applications interactives", *International Eurographics Workshop on Design, Specifaction and Verification of Interactive Systems*, Bocca di Magra (La Spezia).

Bui, T. et Jarke, M, 1986, "Communications Design for Co-op: A Group Decision Support System", *ACM Transactions on Office Information Systems*, Vol. 4, N°2, pp. 81-103.

- Bui, T., 1994, "Evaluating Negotiation Support Systems: A Conceptualization", *Proceedings of the 27th Hawaii International Conference on Systems Science*, Janvier.
- Bui, T., Ma, P-C et Stricker, C., 1995, "Supporting Argumentation in Software Development", working paper, Hong Kong University of Science and Technology.
- Bui, T. et Shakun, M.F., 1995, "Negotiation Processes, Evolutionary systems Design and NEGOTIATOR", *Group Decision and Negotiation*, à paraître.
- Chang, A-M et Han, T-D, 1995, " Design of an Argumentation-Based Negotiation Support System", *Proceedings of the 28th Hawaii International Conference on Systems Science*, Janvier.
- Chang, M-K et Woo C., 1994, "A Speech-Act-Based Negotiation Protocol: Design, Implementation and Test Use", *ACM Transactions on Information Systems*, Vol. 12, N°4, pp. 360-382.
- Churchman, C.W., Ackoff, R.L. et Arnoff, E.L., 1957, *Introduction to Operations Research*, John Wilhey, New-York.
- Conklin, J. et Begeman, M.L., 1989, "gIBIS : A Tool for All Reasons", *Journal of the American Society for Information Science*, Vol. 40, N°3, pp. 200-213.
- Courbon, J-C., 1993, *Systèmes d'information : Structuration, modélisation et communication*, Chap. 5, InterEdition, Paris.
- de Linares, B., Janiaux, P., Reynier, V., Théodule M-L et Werle M., 1996, "Dossier : Naviguer dans le datawarehouse", *Informatiques*, N°16, pp. 82-97.
- Dogson, M., 1993, "Organizational learning : A review of some literatures", *Organization Studies*, Vol. 14, N°3, pp. 375-394.
- El Sawy, O.A., Gomes, G.M. et Gonzales, M.V., 1986, "Preserving Institutional Memory: The Management of History as an Organizational Resource", *Academy of Management Best Paper Proceedings*, Vol. 37, pp. 118-122.
- Fiol, C.M. et Lyles, M.A., 1985, "Organizational learning", *Academy of Management Review*, Vol. 10, N°4, pp. 803-813.
- Haken, H., 1983, *Sybernetics : An Introduction*, 3ème edition, Springer-Verlag, Berlin.
- Hedberg, 1981, How organizations learn and unlearn, P.C. Nystrom et W.H. Starbuck (eds), *Handbook of Organizational Design*, vol.1, Oxford University Press, Oxford.
- Huber, G.P., 1990, A theory of the effects of advanced information technologies on organizational design, intelligence and decision making, *Academy of Management Review*, Vol. 15, N°1, pp. 47-71.



Huber, G.P., 1991, "Organizational learning : The contributing processes and the literatures", *Organization science*, Vol. 2, N°1, pp. 88-115.

Hutchison, A., 1995, "Security in group applications: Lotus Notes as case study", *Information Security - The new decade*, Elhoff et Van Solms (Eds), Chapman et Hall.

Kersten, G.E., 1995, "Simulation and Analysis of Negotiation Processes : The Case of Softwood Lumber Negotiations", *Proceedings of the 28th Hawaii International Conference on Systems Science*, Janvier.

Kersten, G.E. et Cray, D., 1994, "Perspectives on Representation and Analysis of Negotiations : Towards Cognitive Support Systems", working paper, Carlton University.

Kersten, G.E., Michalowski, W., Szpakowicz, S. et Koperczak, Z., 1991, "Restructurable Representations of Negotiation", *Management Science*, Vol. 37, N°10, pp 1269-1290.

Kohonen, T., 1984, *Self-organization and Associative Memory*, Springer-Verlag, Berlin.

Lee, J., 1990, "SIBYL: A Tool for Managing Group Decision Rationales", *Proceedings of Conference on Computer-Supported Cooperative Work*, Los Angeles; 7-10 oct., ACM Press, pp. 79-92.

Liang, T-P., 1993, "Analogical reasoning and case-based learning in model management systems", *Decision Support Systems*, N°10, pp. 137-160.

Martz, B., 1991, "GroupSystems 4.0 : An Electronic Meeting System", *Proceedings of the 24th Hawaii International Conference on Systems Science*.

Meinadier, J.P., 1991, *L'interface utilisateur: Pour une informatique plus conviviale*, Dunod (eds), Paris.

Morrison, J., 1993, "Team Memory : Information Support for Business Teams", *Proceedings of the 26th Hawaii International Conference on Systems Science*, Janvier.

Morrison, J., 1995, "Organizational Memory Information Systems", working paper, Univeristy of Wisconsin, Eau Claire.

Morrison, J. et Weiser, M., 1996, "A Research Framework for Empirical Studies in Organizational Memory", *Proceedings of the 29th Hawaii International Conference on Systems Science*, Janvier.

Nicolis, G. et Prigogine, I., 1977, *Self-Organization in Noneuilibrium Systems: From Dissipative Structures to Order through Fluctuations*, A Whiley-Interscience Publication, John Whiley and Sons, New-York, N.Y.

- Nunamaker, J.F., Alan, R.D., Valacich, J.S. et Vogel, D.R., 1991, "Information technology for negotiating groups: generating options for mutual gain", *Management Science*, Vol. 37, N°10, pp. 1325-1346.
- Pao, Y-H, 1989, *Adaptive Pattern Recognition and Neural Networks*, Addison-Wesley Publishing Company, Inc., Reading, MA.
- Petout, I. et Pigneur, Y., 1989, "An automatic and Visual Approach for User Interface Design", *Proceedings of the 4th IFIP Working Conference on User Interfaces*, Nappa Valley.
- Quinn, R.E. et Rohrbaugh, J., 1983, "A Spatial Model of Effectiveness Criteria : Towards a Competing Values Approach to Organizational Analysis", *Management Science*, Vol. 29, N° 3, pp. 363-377.
- Rao, V.S. et Goldman-Segall, R., 1995, "Capturing Stories in Organizational Memory Systems : The Role of Multimedia", *Proceedings of the 28th Hawaii International Conference on Systems Science*, Janvier.
- Rosen, R., 1985, *Anticipatory Systems : Philosophical, Mathematical and Methodological Foundations*, Pergamon Press, Oxford.
- Sandoe, K., Olfman, L. et Mandviwalla, M., 1991, "Meeting in Time: Recording the Workgroup Conversation", *Proceedings of the 12th International Conference on Systems Science*, New York, décembre.
- Schatz, B.R., 1992, "Building an Electronic Community System", *Journal of Management Information Systems*, Vol.8, N°3, pp. 87-107.
- Schein, E.H., 1984, "Coming to a new awareness of organizational culture", *Sloan Management Review*, N°25, pp. 3-16.
- Schwabe, G., 1994, "Providing for Organizational Memory in Computer-Supported Meetings", *Proceedings of the 27th Hawaii International Conference on Systems Science*, Janvier.
- Shakun, M.F., 1991, "Airline buyout: Evolutionary systems design and problem restructurating in group decision and negotiation", *Management Science*, Vol. 37, N°10, pp. 1291-1303.
- Sougné, J., 1994, "La mémoire", *Cours de psychologie cognitive*, Facultés Universitaires Notre-Dame de la Paix, Namur.
- Spoljar, P., 1995, *Delphi pour Windows*, Sybex (eds), Paris.
- Stein, E.W. et Zwass, R., 1995, "Actualizing Organizational Memory With Information Systems", *Information Systems Research*, Vol. 6, N°2, pp. 85-117.



Sycara, K. P., 1991, "Problem restructuring in negotiation", *Management Science*, Vol. 37, N°10, pp. 1248-1267.

Sycara, K. P., 1993, "Machine learning for intelligent support of conflict resolution", *Decision Support Systems*, Vol. 10, pp. 121-136.

Thomas, K.W., "Conflict and Negotiation Processes in Organizations", *Handbook of Industrial and Organizational Psychology*, Dunnette M.D. (eds), 2ème édition, Vol.3, Consulting Psychologists Press: Palo Alto, CA.

Toulmin, S.E., 1958, *The Use of Arguments*, Cambridge University Press, Cambridge, Mass.

Touretzky, D.S., 1989, *Advances in Neural Information Processing Systems 1*, Morgan Kaufman Publishers, San Mateo, CA.

Tuomi, I., 1995, "Abstraction and History - From Institutional Amnesia to Organizational Memory" , *Proceedings of the 28th Hawaii International Conference on Systems Science*, Janvier.

Walsh, J.P. et Ungson, G.R., 1991, "Organizational Memory", *Academy of Management Review*, Vol. 16, N°1, pp. 57-91.

Watson, R.T., 1996, *Data Management : an organizational perspective*, chap. , John Wiley and Sons.

Wiener, N., 1961, *Cybernetics, or control and communication in the animal and the machine*, 2ème édition, M.I.T. Press, Cambridge, MA.

Winston, P.H., 1984, *Artificial Intelligence*, Addison Wesley, Reading, Mass.

